University of Alberta

DESIGN AND EVALUATION OF A VARIABLE-CAPACITY MULTILEVEL DRAM TEST CHIP

by

Sue Ann Ung

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**.

Department of Electrical and Computer Engineering

Edmonton, Alberta
Spring 2004

Library and Archives Canada

Published Heritage Branch

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque et Archives Canada

Direction du Patrimoine de l'édition

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence
ISBN: 0-612-96558-9
Our file Notre référence
ISBN: 0-612-96558-9

The author has granted a non-exclusive license allowing the Library and Archives Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou aturement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.

Canada

Abstract

As an integral part of electronic digital systems, semiconductor memories play an important role in the development rate and improvement trends of semiconductor technology. Multilevel DRAM (MLDRAM) is one of the ways to increase the storage density by storing more than one bit per memory cell. Several multilevel DRAM schemes have been proposed throughout the history of DRAM research. At the University of Alberta, several previous MLDRAM test chips have been designed and tested functionally. The next step is to characterize a variable-capacity multilevel DRAM chip. ML6, which is an MLDRAM chip built specifically for characterization, is the result of this thesis research work. Some features included in the characterization test chip are: multiple cell sizes, multiple sense amplifier sizes, A/D voltage probes, built-in temperature probes, analog databus set circuits and multilevel sensing operations. The number of levels is adjustable from two-level to three-level, four-level, five-level and six-level. The test chip was proven to function for all levels of operation under normal ambient temperature conditions. Further testing and characterization of ML6 will provide valuable and precise information on the operations of multilevel DRAMs. Once the behavior of multilevel DRAMs is better understood, multilevel DRAMs could become competitive against conventional DRAMs.

The Road Less Travelled

Two roads diverge in a yellow wood,
And sorry I could not travel both
And be one traveller, long I stood
And looked down one as far as I could
To where it bent in the undergrowth;

Then took the other, as just as fair,
And having perhaps the better claim,
Because it was grassy and wanted wear;
Though as for that the passing there
Had worn them really about the same,

And both that morning equally lay
In leaves no step had trodden black,
Oh, I kept the first for another day!
Yet knowing how way leads on to way,
I doubted if I should ever come back.

I shall be telling this with a sigh
Somewhere ages and ages hence:
Two roads diverged in a wood, and I –
I took the one less travelled by,
And that has made all the difference.

– by Robert Frost

To my parents;
my sisters, Jo and Kim;
my Canadian Mom, Angie;
and my favourite Eskimo, John – without whose help, encouragement and great organizational skills, my thesis won't be a reality.

Also, to my friends, who helped me move to Edmonton to attend the U of A.

Acknowledgements

This research was funded by Micronet R&D, MOSAID Technologies Inc., ATMOS Corporation, the Canadian Microelectronics Corporation, the Natural Sciences and Engineering Research Council of Canada, and the University of Alberta.

I would like to thank my supervisors, Dr. Duncan Elliott and Dr. Bruce Cockburn, for their guidance and patience throughout my M.Sc. research program. I would also like to thank my colleagues for their assistance and support: Dan Leder, Tyler Brandon, Michael Redeker, Michael Hume, Craig Joly, John Koob, Christian Giasson and Laleh Najafizadeh.

Table of Contents

Nomenclature

1 Introduction **1**
- 1.1 DRAM Fundamentals . . . 3
- 1.2 Multilevel DRAM Fundamentals . . . 10
- 1.3 Thesis Outline . . . 13

2 Previous Work **15**
- 2.1 Multilevel DRAM Circuits and Techniques . . . 16
  - 2.1.1 Multilevel DRAM Sense and Restore Schemes . . . 19
- 2.2 Multilevel DRAM Chips from the University of Alberta . . . 37
  - 2.2.1 Birk's Four-level MLDRAM . . . 37
  - 2.2.2 Chan's Implementation of Birk's Four-level MLDRAM . . . 43
  - 2.2.3 Xiang's Variable-capacity MLDRAM . . . 50

3 The Design of ML6 **56**
- 3.1 Test Chip Design Overview . . . 56
- 3.2 The Multilevel DRAM Array . . . 60
  - 3.2.1 The Basic Memory Cell . . . 61
  - 3.2.2 The Reference and Generate Cells . . . 66
  - 3.2.3 The Bitline Sense Amplifier . . . 71
  - 3.2.4 The Databus and Read-Write Sense Amplifier . . . 74
  - 3.2.5 The Signal and Wordline Boost Drivers . . . 74
- 3.3 The Peripheral Circuitry . . . 76
  - 3.3.1 Address and Reference-Generate Signal Decoders . . . 77
- 3.4 Built-in Designed-for-Characterization Circuits . . . 79
  - 3.4.1 Databus Analog-Voltage Set Circuit . . . 79
  - 3.4.2 Analog-to-Digital Probe Circuit . . . 80
  - 3.4.3 Temperature Probe Circuit . . . 81

4 ML6 Specifications and Chip Simulation **83**
- 4.1 Design Overview and Specifications . . . 83
- 4.2 The Testing Environment . . . 85
- 4.3 Chip Simulation . . . 86

4.3.1 Analog Chip Simulation with Reduced Core 87

5 Test Chip Debug and Evaluation **95**
5.1 Debug History 96
5.1.1 Databus Functionality 96
5.1.2 Sense Amplifier Functionality 98
5.1.3 Memory Cell Functionality 100
5.1.4 Two-level (DRAM) Operation 105
5.1.5 Three-level Operation 109
5.1.6 Four-level Operation 113
5.1.7 Five-level Operation 116
5.1.8 Six-level Operation 116
5.2 Test Results and Bitmaps 121
5.2.1 Two-level Operation 122
5.2.2 Three-level Operation 123
5.2.3 Four-level Operation 124
5.2.4 Five-level Operation 124
5.2.5 Six-level Operation 125
5.2.6 Cell Yield Comparison 128
5.2.7 Temperature Probe 131
5.3 Discussion 131
5.4 Other Relevant Tests 133
5.5 Conclusion 134

6 Conclusions **136**
6.1 Extracted Layout Simulation versus Actual Chip 136
6.2 Chip Evaluation 137
6.3 Suggestions for Improvement 138
6.4 Accomplishments in this Thesis Research 139
6.5 Future Work 142

Bibliography **144**

A ML6 Pinlist, Tester Connections and Pin Description **149**

B ML6 PGA68 Pin Bonding and Layout Diagrams **154**

C ML6 die **155**

D Sample Output from Visual C++ Code — tester output manipulation **156**

E Probe Point Table **158**

F Row Access Waveforms - the RGX Waveforms **160**

List of Figures

1.1 A conventional 1T-1C DRAM cell 4
1.2 Back-to-back inverters 6
1.3 Sense amplifier circuit 6
1.4 Bitline charge coupling effects on: (a) Folded bitlines (b) Twisted bitlines [6, 20] 8
1.5 Bitline twisting schemes in DRAMs: (a) No twist (b) Single standard twist (c) Triple standard twist (d) Complex twist (e) Single modified twist (f) Triple modified twist [20] 9
1.6 Bitline architectures: (a) Open bitline (b) Folded bitline [6, 20, 42] . 10
1.7 Two-bits-per-cell DRAM scheme 11

2.1 Charge sharing between bitline and memory cell [20] 18
2.2 Data conversion function for Furuyama's two-bit-per-cell MLDRAM [13] 21
2.3 Sub-bitline components of Furuyama's MLDRAM [13] 22
2.4 Read operation in Furuyama's MLDRAM [13] 23
2.5 Write operation in Furuyama's MLDRAM [13] 24
2.6 Schematic of Gillingham's MLDRAM showing four sub-bitlines [14] 26
2.7 Reference and data voltage levels for Gillingham's MLDRAM . . . 27
2.8 MSB sensing steps in Gillingham's MLDRAM 29
2.9 LSB sensing steps in Gillingham's MLDRAM 31
2.10 Data restore steps in Gillingham's MLDRAM 33
2.11 Schematic of Okuda's MLDRAM showing sub-bitline pairs [23] . . 35
2.12 Okuda's MLDRAM bitline hierarchy and time-sharing sense amplifier architecture [23] 37
2.13 "Reference" and "generate" sub-bitline types for Birk's MLDRAM design [5] 39
2.14 Schematic of Birk's MLDRAM scheme showing nine sub-bitline pairs [5] 39
2.15 Birk's sub-bitline connections and reference-generate wordline connections for the reference generation operation [8, 42] 41
2.16 Birk's sub-bitline connections and reference-generate wordline connections for the sensing operation [8, 42] 42

2.17 Birk's sub-bitline connections and reference-generate wordline connections before the completion of the restore operation [8, 42] . . . 43
2.18 Chan's ML3 test chip floorplan [8] 44
2.19 Simplified block diagram of ML3 [8] 46
2.20 Chan's ML3 memory core floorplan showing sub-bitline connections between sections [8] 47
2.21 Wordline boost driver used in ML3 [8] 48
2.22 Row and column address bits in ML3 49
2.23 VDC reference generation voltage sources 51
2.24 ML5 test chip floorplan 52
2.25 ML5 test chip simplified block diagram 52
2.26 Wordline driver from ATMOS Corporation [42] 53
2.27 Row and column addressing bits in ML5 54

3.1 ML6 test chip floorplan 60
3.2 Simplified ML6 test chip block diagram 60
3.3 ML6 memory array showing the sections with five bitline pairs, switch matrix and columns 62
3.4 ML5 memory cell [43] 63
3.5 Schematic of the 1T-1C basic memory cell in ML6 63
3.6 ML6 memory array showing cell arrangement and sizes 64
3.7 Shielding in between sub-bitlines in ML6 65
3.8 ML6 sub-bitline types and arrangement 67
3.9 ML6 switch matrix analog voltages for reference generation 68
3.10 ML6 sense amplifier circuit 73
3.11 Back-to-back inverters 73
3.12 ML6 databus read-write sense amplifier circuitry 75
3.13 ML5 wordline driver from ATMOS Corporation [42] 76
3.14 ML6 wordline boost driver 76
3.15 ML6 X and Y address fields bits 78
3.16 ML6 databus analog-voltage set circuit 80
3.17 ML6 analog-to-digital probe 81
3.18 ML6 temperature probe 82

4.1 ML6 reference voltages and thermometer code for six-level operation 84
4.2 ML6 bitline connections during cell-dump before and after sensing . 84
4.3 ML6 bitline connections during reference generation 85
4.4 ML6 bitline connections during the restore operation 86
4.5 ML6 testing algorithm 89
4.6 ML6 input waveforms going into the memory core 91
4.7 ML6 output and bitline waveforms for input = '00000' 92
4.8 ML6 output and bitline waveforms for input = '10000' 92
4.9 ML6 output and bitline waveforms for input = '11000' 93
4.10 ML6 output and bitline waveforms for input = '11100' 93

4.11 ML6 output and bitline waveforms for input = '11110' 94
4.12 ML6 output and bitline waveforms for input = '11111' 94

5.1 Databus and bitline circuitry 97
5.2 ML6 databus debug input waveforms 99
5.3 ML6 databus debug output waveforms 100
5.4 ML6 sense amplifier debug input waveforms 101
5.5 ML6 sense amplifier debug output waveforms 102
5.6 ML6 cell debug input waveforms 103
5.7 ML6 cell debug output waveforms 104
5.8 Waveforms used to 'delete' the content of memory cells before a write operation 104
5.9 ML6 two-level operation write waveforms 106
5.10 ML6 two-level operation read waveforms 107
5.11 ML6 three-level operation write waveforms 111
5.12 ML6 three-level operation read waveforms 112
5.13 ML6 four-level operation write waveforms 114
5.14 ML6 four-level operation read waveforms 115
5.15 ML6 five-level operation write waveforms 117
5.16 ML6 five-level operation read waveforms 118
5.17 ML6 six-level operation write waveforms 119
5.18 ML6 six-level operation read waveforms 120
5.19 ML6 test algorithm 122
5.20 ML6 six-level operation bitmap for writing thermometer codes to cells in section A 127
5.21 ML6 yield averaged over all cells 128
5.22 ML6 cell yield for the different memory cell sizes 128
5.23 ML6 cell yield for the different sense amplifier sizes for chip #1 . . 129
5.24 ML6 cell yield for the different sense amplifier sizes for chip #2 . . 130
5.25 ML6 cell yield for the shielded and non-shielded bitlines in chip #1 130
5.26 ML6 cell yield for the shielded and non-shielded bitlines in chip #2 131

B.1 PGA68 layout and pin bonding 154

C.1 ML6 die showing die pads, the core and the periphery 155

List of Tables

1.1 Cell Capacities for Various Numbers of Data Levels 12

3.1 ML6 Multiplexed Address Control Opcodes 77

4.1 Power Supplies in ML6 87
4.2 ML6 Six-level Operation Codeword and Voltage Representations . . 89

5.1 Chip #1 Yield Matrix for Two-level Operation 123
5.2 Chip #2 Yield Matrix for Two-level Operation 123
5.3 Chip #1 Yield Matrix for Three-level Operation 123
5.4 Chip #2 Yield Matrix for Three-level Operation 123
5.5 Chip #1 Yield Matrix for Four-level Operation 124
5.6 Chip #2 Yield Matrix for Four-level Operation 124
5.7 Chip #1 Yield Matrix for Five-level Operation 125
5.8 Chip #2 Yield Matrix for Five-level Operation 125
5.9 Chip #1 Yield Matrix for Six-level Operation 126
5.10 Chip #2 Yield Matrix for Six-level Operation 126
5.11 Core Temperature Probe at Room Temperature 131
5.12 Summary of Test Effort 135

A.1 ML6 Pinlist and Tester Connections 149

E.1 ML6 Probe Points 158

List of Nomenclature

Acronyms

1T-1C	One-transistor-one-capacitor DRAM cell design, page 1
3T	Three-transistor DRAM cell design, page 1
4T	Four-transistor DRAM cell design, page 1
A/D	Analog-to-Digital, page 133
BJT	Bipolar junction transistor, page 81
CMC	Canadian Microelectronics Corporation, page 48
CMOS	Complementary Metal Oxide Semiconductor, page 2
DDR-SDRAM	Double Data Rate SDRAM, page 2
DRAM	Dynamic Random Access Memory, page 1
DRC	Design Rule Check, page 95
EDO	Extended Data Out, page 2
FPM	Fast Page Mode, page 2
GALPAT	Galloping One's and Zeros Pattern, page 121
GUI	Graphical User Interface, page 95
HDRAM	An embedded DRAM design using a pure logic process from MOSAID Technologies Inc., page 61
IC	Integrated Circuit, page 85
IO	Input/output, page 45
LSB	Least Significant Bit, page 25
LVS	Layout-versus-Schematic, page 95

MLDRAM	Multilevel DRAM, page 3
MOS	Metal Oxide Semiconductor, page 2
MSB	Most Significant Bit, page 25
NMOS	N-doped Metal Oxide Semiconductor, page 61
PERL	Practical Extraction Report Language, page 122
RDRAM	Rambus DRAM, page 2
SDRAM	Synchronous DRAM, page 2
SPICE	Simulation Program with Integrated Circuit Emphasis, page 59
SRAM	Static Random Access Memory, page 1
TSMC	Taiwan Semiconductor Microelectronics Corporation, page 48
VLSI	Very Large Scale Integration, page 137

Chapter 1

Introduction

Semiconductor memory is an integral part of electronic digital systems. Memories are required to provide the data storage and retrieval function needed for high speed computation. Cache memory, especially, enables fast temporary data storage and retrieval since this memory invariably resides on the same semiconductor chip as the processing unit. Cache memory is typically implemented using static random access memory (SRAM). SRAMs are faster and do not need to be refreshed. However, due to the high cost of SRAMs, only the most expensive and high-end computers have caches any larger than 256 Mb. A more cost-effective approach is to combine caches with slower but much larger off-chip main memories constructed using dynamic random access memory (DRAM). DRAM cells are smaller and higher density than SRAM cells, and are thus cheaper to build [20, 25].

DRAMs make good candidates for main memories because of their low cost per bit and high density advantage. The first DRAMs used four-transistor (4T) and three-transistor (3T) cell designs, which were quickly replaced by the even simpler and hence denser one-transistor-one-capacitor (1T-1C) designs [18, 20]. The 1T-1C cell designs allow for denser cell arrays, including the use of 3-D trench capacitor or stacked capacitor cell structures [18]. DRAMs of 1-Gb density are now in production [18, 27]. DRAMs have slower access speed than SRAMs and consume standby power as a result of the necessary refresh operations that are required to replenish charge lost due to leakage current. However, with the latest[1] deep sub-

[1]The present minimum process feature size is at 0.03 μm.

micron processes, the leakage currents in SRAMs are becoming significant sources of static power dissipation [15].

Within the DRAM family, there are asynchronous and synchronous technologies. DRAM devices with asynchronous interface timing, such as fast page mode (FPM) and extended data out (EDO) memories, are being replaced by DRAMs with clocked signals at the external interface, such as Rambus DRAMs (RDRAMs), synchronous DRAMs (SDRAMs) and double data rate SDRAMs (DDR-SDRAM) [15, 24]. SDRAM technology has become dominant due to faster synchronous operation, multiple memory banks, pipelined burst modes and high-speed bus architectural enhancements [24].

As processor systems advance and the demand for faster computers with larger memories increases, even higher density fast-access memories are needed. Moore's Law[2] predicts that the density and performance of integrated circuits doubles every 18 months [7]. So far, the prediction has been accurate for DRAM technology for just over three decades. However, metal oxide semiconductor (MOS) technology is about to face serious obstacles to further scaling. Some inherent properties — such as sub-threshold leakage, gate oxide defects, process variability, interconnect density, and narrower noise margins — become serious limitations as the technology scales down to below 100 nm [7, 41]

The history of the MOS technology has shown that every time a scaling barrier is encountered, the barrier has been broken through innovative engineering methods and the ingenuity of researchers. Lithography methods have improved and changed in response to the economic incentives that motivate the breaking of these barriers. Chemical and physical properties of other useful materials, such as copper interconnect and high dielectric constant insulators, are being explored and exploited to extend the fundamental limits of MOS technology [32]. Novel three-dimensional structures, such as the double gate transistor, are being investigated, further pushing MOS technology beyond limits foreseen for traditional planar bulk CMOS. Tech-

[2]Gordon E. Moore is presently the Chairman Emeritus of Intel Corporation. He co-founded Intel in 1968. Moore is known for his "Moore's Law", in which he predicted that the number of transistors in a microprocessor system would continue to double every couple of years [11].

nology improvements specific to DRAM mainly lie in the area of process technology and layout architecture such as the trench cell capacitor, stacked cell capacitor structures, deep n-well technology and retrograde p-well DRAM processes [7, 41].

Another technique that can be considered for increasing the storage density of a DRAM is to use cells with multilevel signalling. In multilevel DRAMs (MLDRAMs), the unit cell or combinations of the unit cell are used to store more than 1 bit per cell [5, 42]. This is achieved by having more than two equally-spaced data levels in between the V_{SS} and V_{DD} operating voltages. With this design idea come the challenges of dealing with leakage currents, circuit balancing and sensing techniques that are required to deal with the reduced noise margins and scaling effects. Among the important scaling effects are short channel effects and increased current leakage.

This thesis explores the design and evaluation of a variable-capacity MLDRAM test chip. The integrated circuit presented in this thesis is an extension of work on an earlier test chip designed by Yunan Xiang et al. [42]. The new chip has additional features, such as voltage and temperature probes, to aid in signal characterization and data collection. Multiple cell sizes and sense amplifier sizes are also provided in the new chip so that their effects on data cell storage and sensing can be studied and analyzed. The chip also has a new five-level operating mode to produce a storage capacity of 2.25 bits per cell (i.e. nine bits per 4 cells). The characterization of a chip is the act of running experiments on the chip across different voltages, frequency and temperature. The purpose of characterizing a semiconductor circuit is so that the internal operations and behaviour of the circuit under a wide range of operating environments is measured experimentally.

1.1 DRAM Fundamentals

The DRAM 1T-1C memory cell has the most compact structure of all semiconductor memory cells. The basic structure of one cell is shown in Figure 1.1. Memory cells of this kind are arranged in a rectangular array of cells that can be accessed via orthogonal wordlines and bitlines. The wordline (i.e. row) and bitline (i.e. column)

addresses are usually latched and decoded in the peripheral circuitry outside of the DRAM cell array.

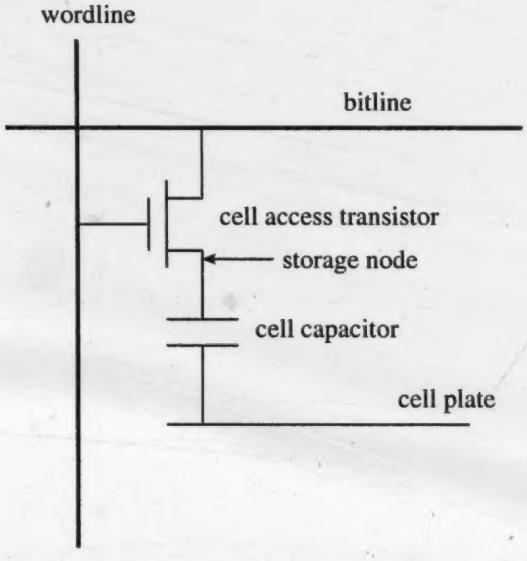


Figure 1.1: A conventional 1T-1C DRAM cell

The capacitor in the basic 1T-1C cell is used to encode the stored data as the voltage of the stored electrical charge. The cell is accessed and the stored charge is shared onto the bitline via the access transistor. The wordline is used to activate the access transistor so that the data signal can be dumped onto the so-called true bitline and the resulting attenuated bitline voltage signal can then be compared with a reference voltage on a second, so-called complement bitline. The comparison is made by a sensitive differential-mode sense amplifier. The voltage difference across the true and complement bitlines is amplified by the sense amplifier to recover the binary cell data. The cell-plate is a common node among all the memory cells. The voltage value of the cell-plate is typically $\frac{1}{2}V_{DD}$. A $\frac{1}{2}V_{DD}$ cell-plate provides the lowest possible stress across the cell dielectric for storing both zeros (V_{SS}) and ones (V_{DD}) [15, 25].

Typically, before sensing (reading) is performed on a DRAM cell, a reference voltage must be generated and dumped onto the complement bitline. This prepares the bitlines so that the sense amplifier can compare the sensed voltage against an ap-

propriate reference voltage when the sensed voltage is available on the true bitline. The reference voltage for a conventional DRAM is $\frac{1}{2}V_{DD}$. For a two-level DRAM operation, the reference voltage of $\frac{1}{2}V_{DD}$ allows for the storage of one bit per cell. Since the precharge voltage for a conventional DRAM happens to be $\frac{1}{2}V_{DD}$, the reference voltage is conveniently generated during the precharge operation.

To write data into a DRAM cell, the true bitline is driven to either V_{DD} or V_{SS}, depending on the data ('0' or '1') to be stored. If the data cell is located on a complement bitline then the opposite signal encoding is used. Next, the wordline is activated to turn on the cell access transistor. With the wordline turned on, the storage node is then charged to the same voltage as the bitline. Finally, the wordline is deactivated to isolate the data signal on the storage node.

To read from a cell, the bitline is again first precharged to $\frac{1}{2}V_{DD}$. Then the wordline is activated to turn on the cell access transistor connecting the storage node to the bitline. This causes the stored charge to be dumped onto the bitline causing a detectable voltage change in the bitline; meanwhile, the complementary bitline stays at the reference voltage of $\frac{1}{2}V_{DD}$. The bitline voltage should now be either slightly (eg. 100 mV) below or above $\frac{1}{2}V_{DD}$ depending on the data stored previously in the cell. The small difference in voltages between the bitline and the complementary bitline is amplified by a sense amplifier. After being amplified by the sense amplifier, a column may be selected and connected to the databus.

The sense amplifier is a simple differential amplifier consisting of two CMOS inverters connected back-to-back in a ring, as shown in Figure 1.2. The true and complement bitlines are shown as BL and BLn, respectively. Before sensing, the sense amplifier power is disconnected and the two sensing nodes are precharged to $\frac{1}{2}V_{DD}$.[3] The sense amplifier is then connected to the bitline pair. The differential mode sense amplifer increases the voltage difference across a bitline pair and drives the bitline signals to opposite voltage rails. The sense amplifier is not re-

[3] A $\frac{1}{2}V_{DD}$ sensing scheme is used in most DRAMs for lower power consumption. When the bitlines and sense amplifiers are precharged to $\frac{1}{2}V_{DD}$, the voltage swing required to put the bitlines at rail-to-rail voltages is less. In a large memory array system, the power consumption that can be reduced in this way is significant.

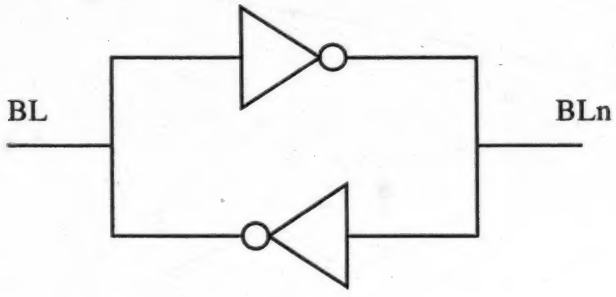


Figure 1.2: Back-to-back inverters

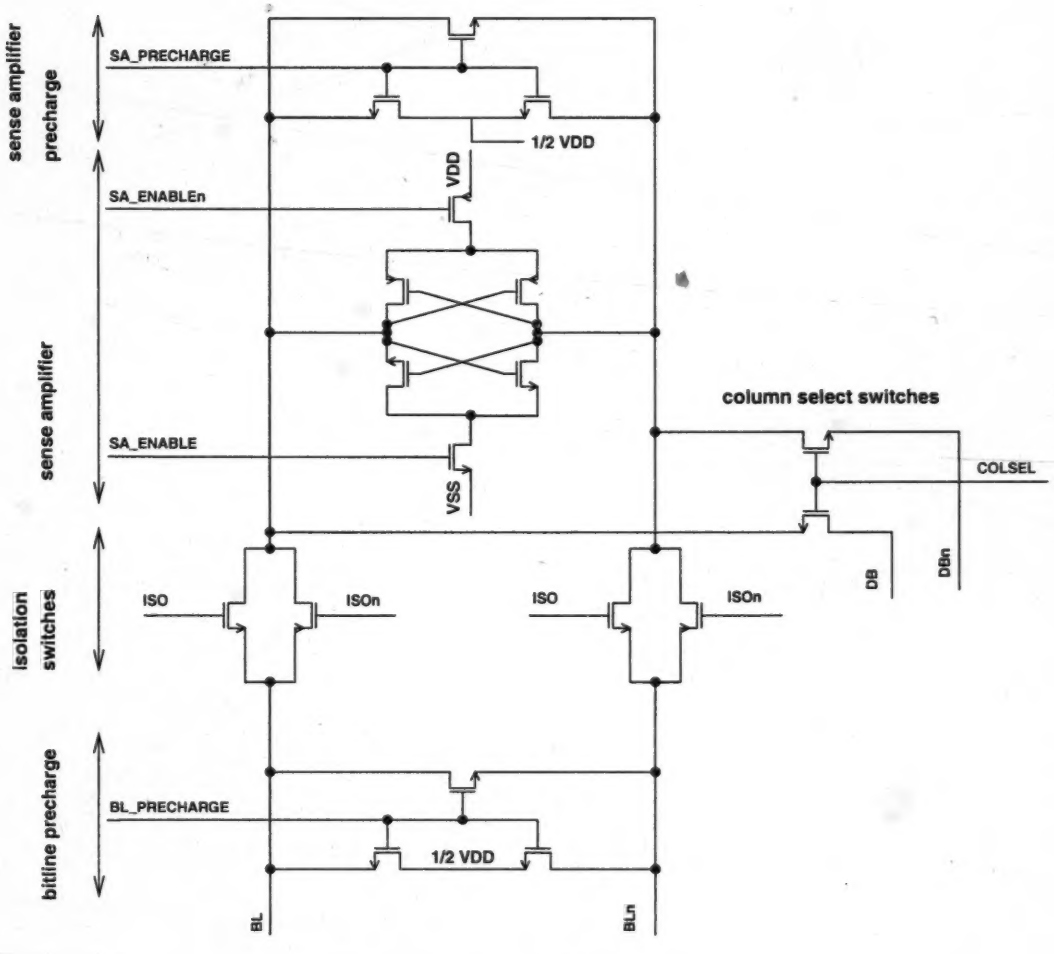


Figure 1.3: Sense amplifier circuit

quired during the write operation. A separate low-impedance write driver circuit in the periphery will usually be used to overdrive the sense amplifier, which will typically be left powered on during the write operation. At the completion of the write operation, the sense amplifier is used to make sure that the bitlines are fully at rail-to-rail voltages before the data charge is isolated in the cell storage capacitor by de-asserting the wordline. Figure 1.3 shows a typical sense amplifier circuit with isolation transistors, precharge devices and select switches. The bitline and sense amplifier configuration shown in Figure 1.3 was used in the new chip for this thesis work. It is a modification from a previous chip designed in [42]. Note that the suffix "n" is used to indicate a complemented or active low signal.

Since accessing the cell involves charge-sharing between the cell capacitance and the precharged bitline capacitance, the ratio of the cell and bitline capacitance is important. In typical DRAM processes, the cell capacitance to bitline capacitance ratio is in the range of 1:8 to 1:10 [20]. A typical cell capacitance would range from 20 fF to 60 fF [18]. The magnitude of the resultant voltage on the bitline after charge-sharing is thus a function of the cell voltage, the bitline and cell capacitances, in addition to the original cell voltage. In conventional DRAMs, the attenuated data signal on the bitline is roughly 100 to 200 mV [18].

Most DRAM bitline architectures employ bitline folding and some form of twisting. These bitline arrangements improve noise immunity [19]. Due to this advantage, the folded array architecture has been used since the 64 kbit generation of DRAMs [21]. When combined with some form of bitline twisting, the signal-to-noise performance improves significantly. The folding and twisting of the bitlines balances the capacitive coupling between adjacent and all other bitlines, improving overall noise immunity in the DRAM array, as depicted in Figure 1.4 [8, 2, 20, 21]. In Figure 1.4, without bitline twisting in a folded bitline architecture, BL_A and BLn_A will experience unequal capacitive disturbances since BLn_A is also affected by BLn_B, while BL_A is not. In a twisted bitline architecture, however, both BL_A and BLn_A (also BL_B and BLn_B) experience the same common mode bitline capacitive coupling disturbances, which are easily rejected by the sense am-

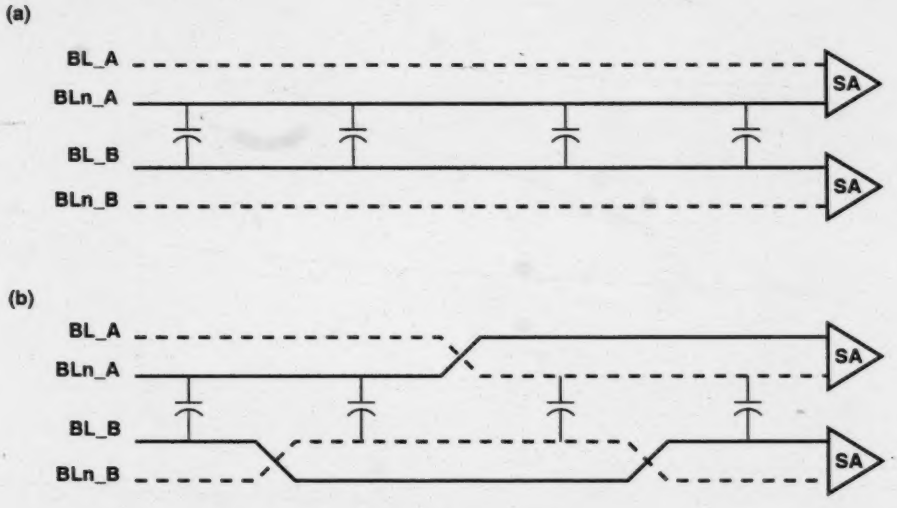


Figure 1.4: Bitline charge coupling effects on: (a) Folded bitlines (b) Twisted bitlines [6, 20]

plifier.

Figure 1.5 shows some conventional bitline twisting schemes that can be employed in DRAM arrays [20, 30]. Without the folded bitline architecture, the sense amplifiers would be connected as shown in Figure 1.6(a) [16, 17, 20]. The open bitline form of DRAM architecture was prevalent in the pre-64 kbit DRAM designs. As the memory arrays get larger, assuring noise immunity posed a bigger challenge than maximizing the array density, therefore the folded bitline architecture with bitline twisting was used. From the figures, it can be seen that the folded bitline architectures forces the bitline array to be larger than the open bitline architecture as shown in Figure 1.6 [20]. In the open bitline architecture, a memory cell is contained in every WL-BL intersection; whereas, in the folded bitline architecture, a memory cell is contained in every other WL-BL intersection [20]. Nevertheless, the folded bitline offer a much greater noise immunity advantage, which is a trade-off that is favoured over the higher storage cell density provided by the open bitline architecture [20].

Another method that is used to improve noise immunity is the use of friendly

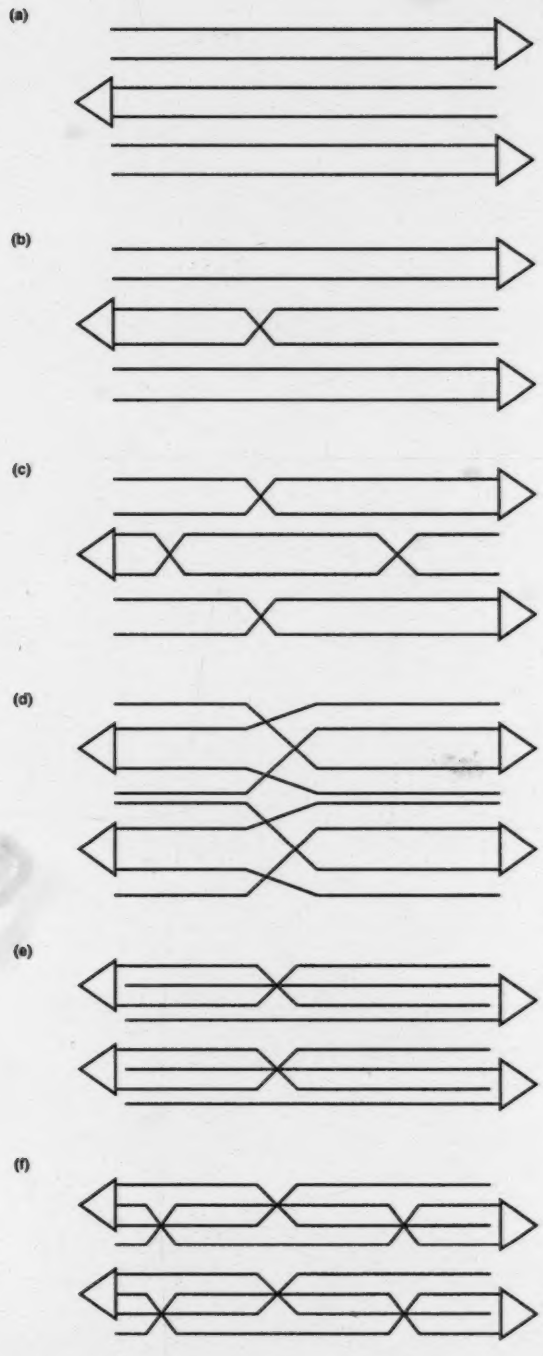


Figure 1.5: Bitline twisting schemes in DRAMs: (a) No twist (b) Single standard twist (c) Triple standard twist (d) Complex twist (e) Single modified twist (f) Triple modified twist [20]

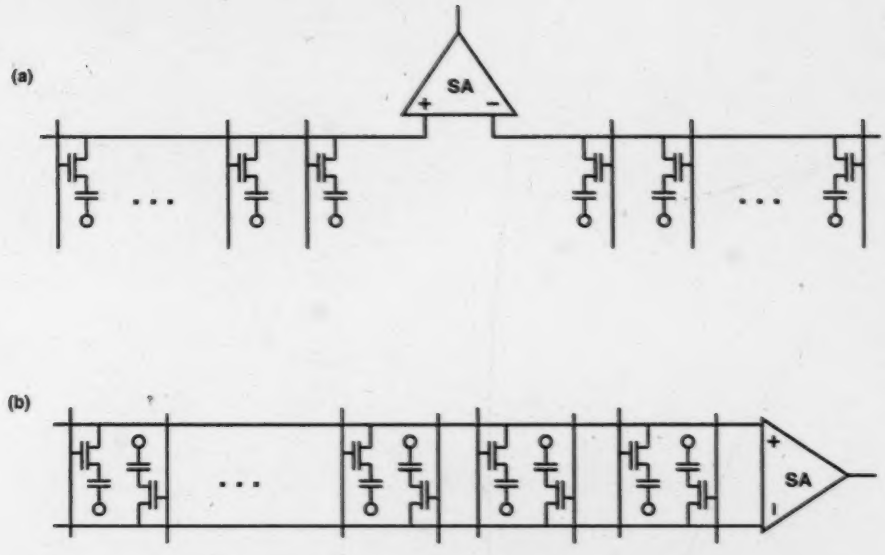


Figure 1.6: Bitline architectures: (a) Open bitline (b) Folded bitline [6, 20, 42]

(dummy[4]) cells. Adding friendly cells at the edges of the memory array provides the edge cells the same physical and electrical environment as any of the other cells in the inner parts of the array [20, 22, 44]. The memory cells at the edges of the array experience a topographical discontinuity that can be avoided with the addition of friendly cells. Friendly cells are also added to the areas immediately adjacent to the bitline twist regions for the same reason. The friendly cells balances the parasitic capacitances in that they enable all memory cells to be surrounded by identical memory cells. The friendly cells are identical to the other memory cells but are deactivated by gate connections to V_{DD} and V_{SS} for PMOS and NMOS transistors, respectively [20]. The friendly cells are deactivated so that they are electrically inate and do not affect the operation of the real cells.

1.2 Multilevel DRAM Fundamentals

Cost has been the most important issue in DRAM designs since DRAM was first introduced about three decades ago. DRAM has always been intended to achieve

[4]Memory cells used to store reference voltages, and for charge balancing and charge injection cancellation during sensing are also sometimes called dummy cells.

the lowest possible cost per bit for a semiconductor memory. With the issue of cost come other issues fundamental to large-scale manufacturing, such as density, yield and access speed. The multilevel technique is just one more way to increase the effective density of DRAM.

In an MLDRAM cell, more than 1 bit is stored in each cell. This is done by increasing the number of reference levels within the V_{SS} to V_{DD} operating range. For example, in a four-level DRAM, the three reference voltages can be set equally-spaced over the supply voltage range to store 2 bits of data in the manner shown in Figure 1.7.

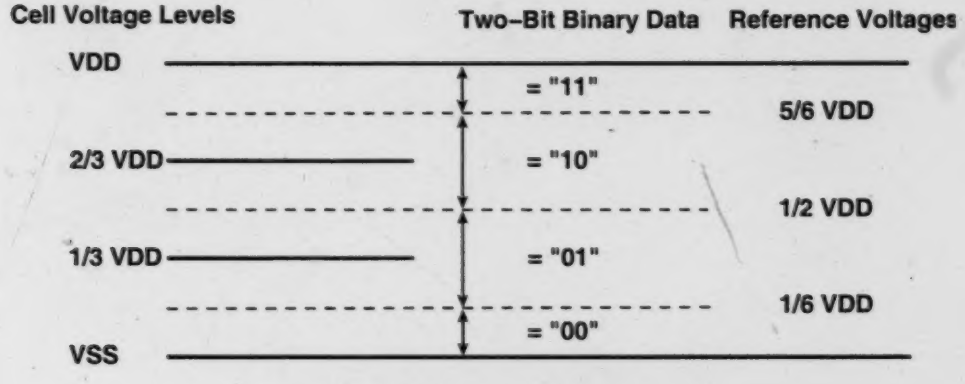


Figure 1.7: Two-bits-per-cell DRAM scheme

Three reference voltages are needed to sense and hence recover a 2-bit data signal. This multi-bit storage idea is expandable to more than 2 bits per cell, as shown in Table 1.1 [5]. The reference voltages are equally spaced in between V_{SS} and V_{DD} in order to maximize the noise margin in between data and reference signals. Equations (1.1), (1.2) and (1.3), taken from [5], were used to make and calculate the entries in the table.

$$V_{cell} \in \{0,1,2,...,N-1\}\frac{V_{DD}}{N-1} \tag{1.1}$$

$$V_{REF} \in \{1,3,5,...,2N-3\}\frac{V_{DD}}{2(N-1)} \tag{1.2}$$

$$n = \frac{1}{Q}\lfloor log_2 N^Q \rfloor \tag{1.3}$$

Table 1.1: Cell Capacities for Various Numbers of Data Levels

Levels, N	Required Ref. Levels, V_{ref}	Cells Encoded, Q	Bit(s)-per-cell[5], n
2	1	1	1
3	2	2	1.5
4	3	1	2
5	4	4	2.25
6	5	2	2.5
7	6	2	2.5
8	7	1	3

For N data levels in a cell, the number of reference levels required for sensing is $N-1$. Equation (1.2) shows the corresponding reference levels for the various data voltage levels from Equation (1.1) [5, 8, 42]. The corresponding cell capacity can be calculated from Equation (1.3) [43]. These equations and the variable-capacity idea are explained in detail in two previous theses that describe the ML3 [5, 8] and ML5 [42] test chips.

A challenge in any MLDRAM is that the available noise margin is reduced as the number of levels increases.[6] By Equation (1.1) and Equation (1.2), the noise margin at 6 levels of operation is $\frac{1}{6}V_{DD}$ compared to the noise margin of $\frac{1}{2}V_{DD}$ in the conventional DRAM. Also, due to the dynamic nature of DRAM, the inherent charge leakages from the cell capacitors due to sub-threshold conduction through the access transistor will be a serious concern because they will limit the retention time between refresh operations. This situation will get even more challenging in future years as the transistor channel length gets shorter and the supply voltage decreases due to scaling of the process technology.

[5] As an example, for $N = 6$, the cells are encoded in pairs. Equation (1.3) yields $n = 2.5$ bits. So, with six-level operation, two cells can be used together to generate five bits.

[6] For a circuit to be robust, it has to be insensitive to noise. The noise margin of a signal is given by the allowable range or voltage separation a signal can vary within before it is interpreted as being a different (i.e. incorrect) digital value.

The sense and restore circuit operation in a MLDRAM is more complicated than in a conventional DRAM. Additional circuits and silicon are required in the periphery, possibly decreasing the speed of writing and sensing, and possibly increasing power consumption. There are two types of sensing schemes: parallel and sequential. In parallel sensing, multiple copies of data signal can be sensed at the same time with multiple references and multiple sense amplifiers [6, 13, 42]. Sequential sensing differs from parallel sensing in the way that the reference voltages are generated. In sequential sensing, the bits in a valid data code are sensed with reference voltages that are determined by the previously sensed bit(s) [1, 14, 23].

The reference generation circuitry in an MLDRAM is also more complicated than in a conventional DRAM because more than one reference level has to be generated and stored. The references can be generated either globally or locally. Globally-generated reference levels are generated from outside of the MLDRAM array while locally-generated reference levels depend on circuitry inside the DRAM cell array for reference generation. Charge-sharing techniques, such as the capacitive coupling between sub-bitlines, are used with built-in switch matrices to perform local reference level generation. The switch matrix connects the bitlines to create the desired reference levels through charge sharing. By connecting the bitlines in a certain way, the sensing circuitry can be balanced capacitively between the reference and data inputs of the sense amplifier. Circuit balance (with respect to signal attenuation and noise coupling) and a sufficiently small bitline-to-cell capacitance ratio are important for noise reduction and correct circuit operation.

1.3 Thesis Outline

The end result of this thesis was a chip designed to facilitate the characterization of the variable-capacity MLDRAM. The goals were to facilitate further investigation of the effectiveness of the reference generation technique, sense amplifier sizes and different bitline-to-cell capacitances in the operation of the MLDRAM. The next chapter of this thesis introduces the different sensing and reference generation schemes that have been used in previously proposed MLDRAM designs. Two pre-

vious MLDRAM chips designed at the University of Alberta are also reviewed in this chapter. Chapter 3 describes the design and CAD simulation of the new characterization chip. Chapter 4 explores the design flows used in the design of the chip and describes the operational specifications and testing. Chip verifications for prototype chips are presented in Chapter 5. Conclusions and proposed future work are described in Chapter 6.

Chapter 2

Previous Work

The conventional DRAM stores one bit per memory cell. For a data cell on a true (complement) bitline, with a reference voltage of $\frac{1}{2}V_{DD}$, a data voltage above $\frac{1}{2}V_{DD}$ represents a '1' ('0') bit while a data voltage below $\frac{1}{2}V_{DD}$ represents a '0' ('1') bit. A single cell in a DRAM array can be made to store more than one bit by increasing the number of data voltages in the range from V_{SS} to V_{DD}. Table 1.1 from Chapter 1 shows how the number of bits stored in a cell can be increased by increasing the number of data levels in a multilevel DRAM.

The core of the multilevel DRAM cell array does not change much from that of a conventional DRAM. The 1T-1C DRAM cell has by far the most compact layout of any memory cell and it makes sense to just re-use a technology that is already proven. In a multilevel DRAM (MLDRAM) scheme, the sensing and reference generation circuitry surrounding the core are crucial to the correct operation of multi-bit sensing and multilevel reference generation.

In an effort to reduce the bit-cost in DRAMs, advances in lithographic technology have enabled chip-size reductions and density increases [23]. In multilevel DRAMs, the challenges of the already narrow noise margins are compounded by the leakage currents and more pronounced capacitive coupling effects that arise due to the reduced dimensions in the transistors and other semiconductor components.

In conventional DRAMs, the differential sense amplifier, the reference generation circuits, and the driver circuits all fit within a bitline pitch of two columns of cells. The circuits are staggered to give a two-column pitch. In MLDRAMs such as

Birk's [5, 8] for example, staggered sense amplifiers cannot be used, so the pitch in Birk's MLDRAM is now one column, and not two. Staggering of circuits, such as the sense amplifier circuits is difficult to achieve in MLDRAMs as the sensing and reference generation circuits are larger and non-repetitive as compared to a conventional DRAM's. Also, with the increase in the number of data levels, the noise margins are reduced. For a conventional two-level DRAM, the noise margins are $\frac{1}{2}V_{DD}$ at the cell, and are further reduced after charge sharing. A four-level DRAM, for example, has a noise margin that is $\frac{1}{3}$ that of a two-level DRAM with the same supply voltage. In general, the noise margins on an N-level MLDRAM are reduced by $\frac{1}{N-1}$ compared to those of a two-level DRAM [5].

The reduced noise margins make the MLDRAM more susceptible to soft errors, leaky cells[1] and sense amplifier offsets. Leakage currents that could be safely tolerated in conventional DRAM designs have become major concerns in MLDRAM designs. These problems are further accentuated with the further scaling of CMOS transistors [41].

There are several sense and restore schemes that have been used for MLDRAM [1, 6, 13, 14, 23, 43]. The sensing can be done sequentially or in parallel with local or global reference voltage generation schemes. These sensing and reference generation schemes have their advantages and disadvantages, and are described in the following sections.

2.1 Multilevel DRAM Circuits and Techniques

The sense and restore circuitry in an MLDRAM must be able to deal with the sensing of multiple (more than two) data signal levels. The multiple sensed signals must then be decoded to represent multi-bit information. The restore circuitry must also be able to convert multi-bit data that is input into one of a few nominal data voltage levels to be stored on the memory cell capacitor.

[1]Leaky cells would cause problems in MLDRAMs because it will induce more noise into the system. The noise margin would be further reduced. Section 2.1 explains MLDRAM circuits and noise margins.

Most of the MLDRAM sensing and reference generation operations use charge sharing operations between the cell and bitline parasitic capacitances to generate the nonstandard intermediate voltages. Figure 2.1 illustrates the charge sharing process. When the wordline is not asserted, the bitline contains the precharged value $\frac{1}{2}V_{DD}$. The cell node, at this point, is at an unknown voltage (probably in between V_{SS} and V_{DD}), unless it is purposefully precharged to $\frac{1}{2}V_{DD}$ before a write sensing operation.[2] When the wordline is asserted, the bitline and the cell node voltages blend to give a new voltage on both the bitline and cell node. The new voltage is a function of the combination of the bitline and cell capacitances. Equation (2.1) gives the relation between the blended voltage V_{signal}, the bitline $C_{bitline}$ and cell node C_{cell} capacitances, and the initial cell voltage V_{cell}.

From conservation of charge, it is straightforward to show that:

$$V_{signal} = \left(V_{cell} - V_{cell-plate}\right)\left(\frac{C_{cell}}{C_{bitline} + C_{cell}}\right) + V_{bitline}. \tag{2.1}$$

Equation (2.1) can be used to predict the resulting voltage on the bitline after charge sharing in multilevel operation environment [20].

In a typical case, where the cell capacitance is $50\,fF$ and the bitline capacitance is $500\,fF$, the voltage appearing on the bitline would be $0.485\,V_{DD}$ if the voltage stored in the cell is $\frac{1}{3}V_{DD}$ assuming four-bits-per-cell operation and assuming a bitline precharge and cell-plate voltage of $\frac{1}{2}V_{DD}$. This would mean that the voltage differential seen by the sense amplifier is $(0.5 - 0.485)\,V_{DD} = 0.015\,V_{DD}$. For 0.18-$\mu$m technology, a typical power supply of $1.8\,V$ will give a voltage differential on the bitline and complement bitline of $27\,mV$. This value is a very small fraction of the differential bitline voltage value typically seen in a conventional DRAM (100 to $200\,mV$).

$$V_{signal} = \left(\frac{1}{3}V_{DD} - \frac{1}{2}V_{DD}\right)\left(\frac{50}{500+50}\right) + \frac{V_{DD}}{2} \tag{2.2}$$

[2] Although uncommon in most DRAM operations, the cell node can be precharged to $\frac{1}{2}V_{DD}$ so as to initialize the cell before writing. This can be done by asserting the addressed wordline for a split second before the precharge enable signal is turned off.

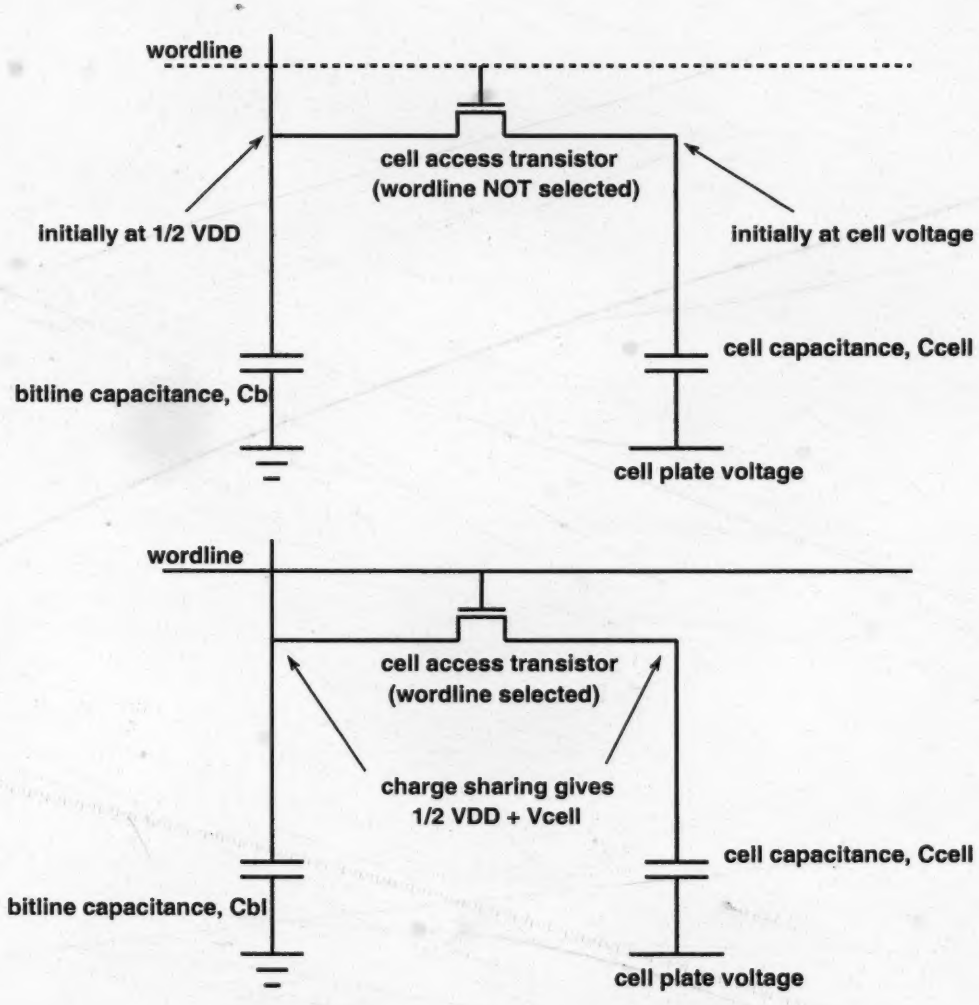


Figure 2.1: Charge sharing between bitline and memory cell [20]

$$= \frac{32}{66} V_{DD}$$
$$= 0.485 V_{DD}$$

During sensing in a conventional DRAM, before data is read from the bitlines, the sense amplifiers are turned off and the bitlines are precharged to $\frac{1}{2}V_{DD}$ and then isolated. Then the wordline is asserted to turn on the cell access transistor so that the charge from the memory cell will be dumped onto the floating bitline. The charge sharing causes the bitline voltage to rise or fall slightly below the precharge voltage. Then the sense amplifier is enabled so that the differential voltages on the bitline and complementary bitline can be amplified and pulled rail-to-rail. This data is then driven as a '1' or '0' over the differential databus to the pads.

For reference signal generation, groups of bitlines are electrically isolated (by opening transistor switches) and then precharged to readily available values such as V_{SS}, V_{DD} or $\frac{1}{2}V_{DD}$. Then they are connected in such a way that the desired nonstandard signal voltages are generated via charge sharing between the bitlines. We will call this method the "switch matrix" or Birk's reference generation method [14].

The charge sharing proces described above is just one of the ways the signals can be generated. Another method, although less desirable because of its sensitivity to process variations, is by capacitive coupling [23].[3] In this method, specific reference voltages are generated by connecting sub-bitlines with built-in capacitors for charge-sharing. The built-in capacitors can cause imbalances in the MLDRAM sensing circuit. This is why this method of reference generation is less likely to be successful in production parts. The following sections detail the sense and restore schemes that have been proposed in the literature so far.

2.1.1 Multilevel DRAM Sense and Restore Schemes

In order for MLDRAMs to be a feasible replacement for the existing two-level DRAM designs, the key internal operations, such as sensing and restoring, have to

[3]The NEC scheme [23] uses two-step sensing, the first sensing using a $\frac{1}{2}V_{DD}$ reference voltage. This step bumps a second floating reference line either higher or lower in voltage. The coupling capacitor is sized so that the bumped voltage is the required second reference voltage — see section 2.1.1.3 on Okuda's MLDRAM.

be analyzed and their strengths and weaknesses addressed.

Some very important criteria have to be met in order to compete with the existing DRAM designs [20, 25]:

Area The sense and restore circuitry size must fit within the bitline pitch. In conventional DRAMs, the sense and restore circuitry fit within the bitline pitch of one or two memory cell columns. The relaxed pitch of two columns is possible if the sense amplifiers can be staggered on opposite sides of the array.

Process variations The MLDRAM circuits should be made robust against expected process variations. Charge sharing operations are especially sensitive to process variations due to the inherent nature of metal line parasitic capacitances in the bitlines and memory cells. Variable metal sizes, dielectrics and spacings within the chip layout are produced by process variations and the variations can affect the quality of the signals in the circuit.

Noise Noise insensitivity is also important when smaller transistor sizes are used, such as those in 0.18 μm or 0.13 μm CMOS technology[4]. Although the reduced operation voltage minimizes the unavoidable leakage current from the memory cells, the noise margin is significantly reduced from the multilevel reference voltages. Also, sense amplifier offsets are more pronounced when the transistor sizes get smaller.

Speed Fast DRAMs such as SDRAMs, Rambus DRAMs and DDR-SDRAMs have surpassed 1 Gigahertz [24]. MLDRAM is significantly slower in access times due to the increased number of operations in the sense and restore scheme. The reference generation operation in MLDRAM is more complicated than in conventional DRAM.

To better understand the problems expected in MLDRAM designs, an essential step is to characterize an MLDRAM test chip. Some of the more promising sens-

[4]Leakage current depends on the operating voltages and the off resistance of the cell access transistors.

ing and restoring schemes have been selected and analyzed. From an analysis of the strengths and weaknesses of the earlier designs, further new test chips can be developed and characterized.

2.1.1.1 Furuyama's MLDRAM — Parallel Sensing and Global Referencing

A two-bit per cell storage MLDRAM is proposed by Furuyama et al. [13] that adapts parallel sensing and a global reference generation scheme. For a two-bit per cell system, three reference levels (V_{DCA}, V_{DCB} and V_{DCC}) in between V_{SS} and the maximum operating voltage V_{DD}, are required to sense two-bit data values. Figure 2.2 shows how the voltage levels can be mapped to data values.

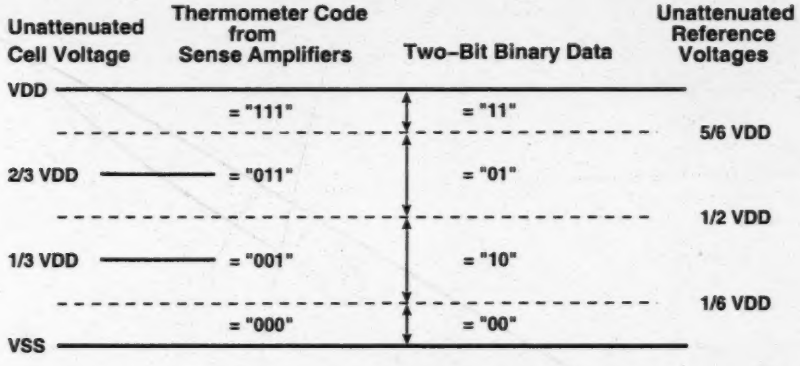


Figure 2.2: Data conversion function for Furuyama's two-bit-per-cell MLDRAM [13]

In global reference generation, the reference voltages are generated by voltage generators external to the MLDRAM circuitry. The reference voltages are then propagated across the MLDRAM circuitry from the external sources. This method of reference generation and distribution may induce degradation in the reference signals due to the variations in the circuit parameters across the MLDRAM circuitry. Signal voltage drops as the signal is distributed through the wiring and long wiring is susceptible to noise pickup. Also, on-chip (but external to the main MLDRAM circuitry) voltage generators are not accurate in generating the reference voltages.[5] This method of reference generation is, nevertheless, simpler and

[5]Reference voltage generators can be realized by a current-mirror amplifier, which may not provide the accuracy needed for the noise-sensitive MLDRAM architectures [13].

sometimes faster than the local reference generation method that is used in the ML-DRAMs designed by Aoki et al. [1], Gillingham [14], Okuda et al. [23] and the University of Alberta [5, 8, 42].

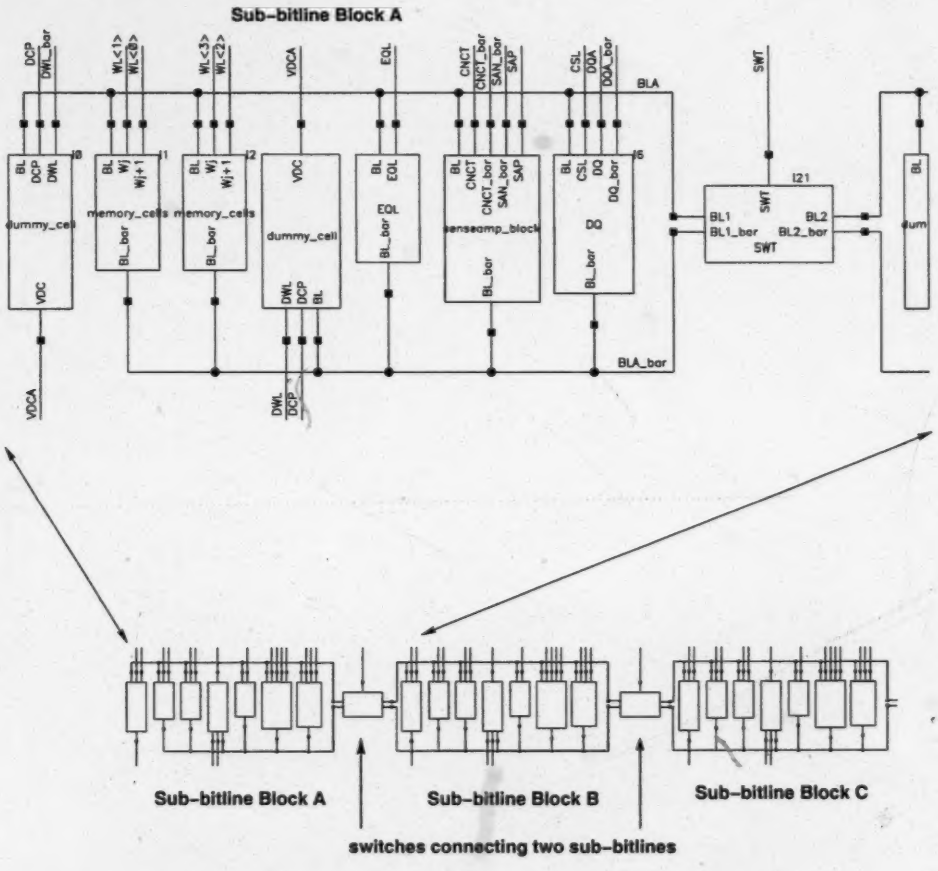


Figure 2.3: Sub-bitline components of Furuyama's MLDRAM [13]

The sensing scheme used in Furuyama's MLDRAM design is called parallel since the comparison with reference voltages occurs in parallel with a different sense amplifier in each block. Figure 2.3 shows how this sensing scheme can be realized. The A, B and C blocks are identical except for the globally generated reference voltages. Each bitline pair (real and complement bitlines) segment in this design is divided into the three blocks. These blocks can be connected or disconnected via switches so that three separate copies of the charge-shared bitline voltage can be compared simultaneously to three different reference levels on the comple-

ment bitlines. Each block is a sub-bitline pair with dummy[6] cells for reference generation, a sense amplifier and memory cells.

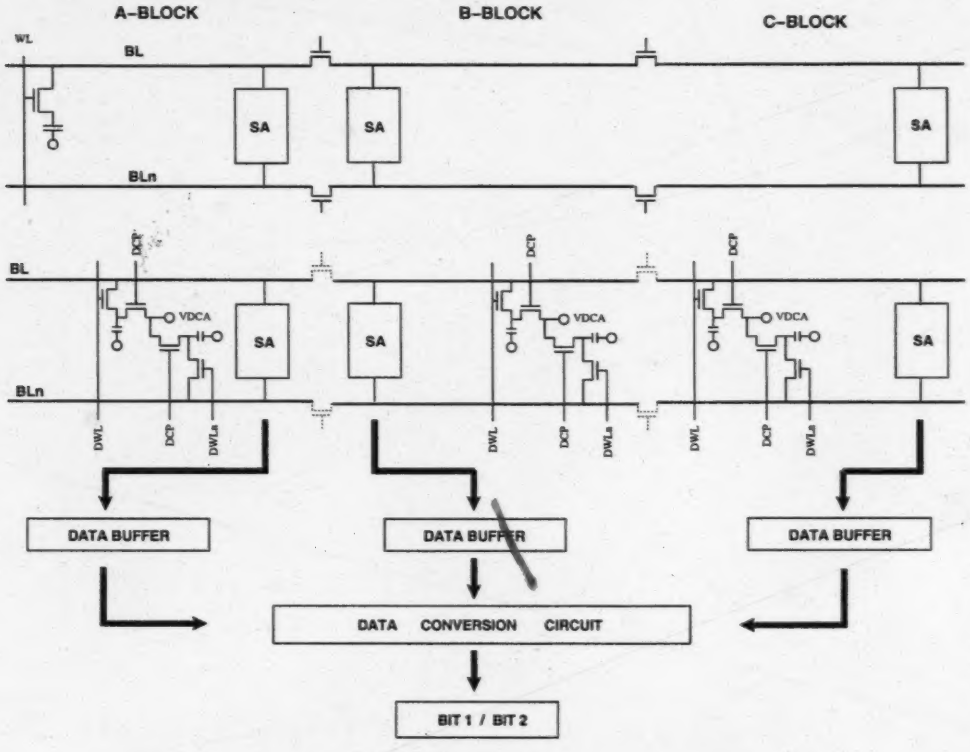


Figure 2.4: Read operation in Furuyama's MLDRAM [13]

Figure 2.4 summarizes the read operation in Furuyama's MLDRAM [13]. The read cycle starts with the bitlines precharged to $\frac{1}{2}V_{DD}$ and the bitline switches in between the sub-bitlines pairs asserted so that the three sub-bitline blocks are connected. Next, the addressed wordline is asserted to dump the contents of a data cell onto the real bitline while the complement bitline remains at the precharge voltage of $\frac{1}{2}V_{DD}$. Then the bitline switches are turned off to disconnect the sub-bitline blocks. Following this, the dummy wordlines are asserted so that the three reference values are transferred from the dummy cells onto the complement sub-bitlines. In each of the separated sub-bitline blocks, the data voltage value is compared with the reference voltage value by the sense amplifier. The resulting amplified data voltage

[6]In Furuyama's MLDRAM, dummy cells are used to store reference voltages that are generated from outside of the MLDRAM memory circuit.

value from each of the real sub-bitlines is then driven onto the output buses to be decoded according to the data conversion diagram shown in Figure 2.2.

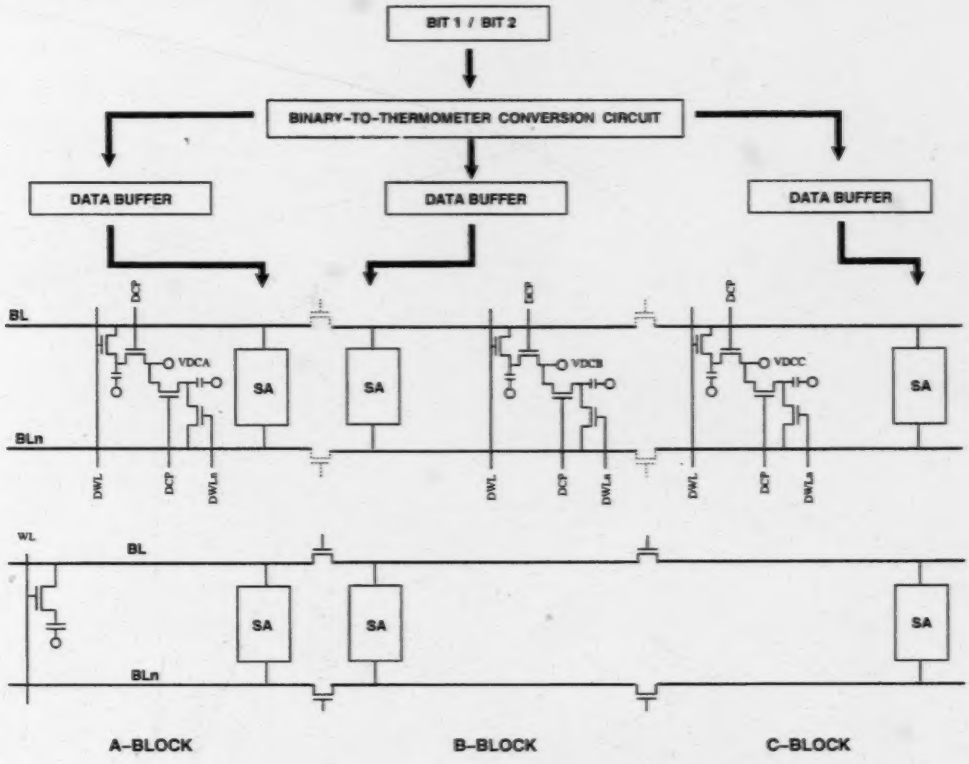


Figure 2.5: Write operation in Furuyama's MLDRAM [13]

Figure 2.5 explains the write operation for Furuyama's MLDRAM [13]. The two-bit input data code is first encoded into three bits according to the same data conversion table that was used for reading. At this point, the bitline switches are deasserted so that the bitline pair is separated into the three sub-bitlines blocks. Then, the three bits are written one bit each into the real sub-bitlines with the sense amplifier in each block activated. Next, the sense amplifiers are disabled and the resulting three floating sub-bitline blocks are connected so that the values on the real sub-bitlines are blended together to get a new value on the full bitline. Charge sharing along the full bitline produces the correct new data voltage value. This voltage value is then stored in a cell when the addressed wordline is deasserted. The restore operation after a read can be done in steps similar to the write operation

as described before.

The single-step parallel sensing method in this MLDRAM design makes the read and write operations fast compared to other MLDRAM designs.[7] The three sense amplifiers are used to recover the bits simultaneously. However, the requirement of one sense amplifier per sub-bitline block increases the area per bit for the MLDRAM. The extra circuitry needed to decode and convert the bits also adds to the overall area.

Furuyama's design uses globally-generated reference voltages. Reference voltage generation is done outside of the MLDRAM circuitry and involves the distribution of analog voltages across the memory array to reach all the sub-bitlines. Globally generated reference voltages are subject to voltage drops and coupled noise as they propagate through the memory array. This problem is inherent to globally generated reference signals and poses a major disadvantage to such an MLDRAM scheme.

In [13], Furuyama et al. suggest using a trench capacitor process technology to increase the storage cell capacitance without having to increase the area of the cell. Small capacity cells may not give a strong enough voltage on the bitline after charge sharing. The weak cell signals will in turn cause poor retention times, bad soft error immunity and reduced noise margins [13]. Furuyama et al. also stress the importance of having balanced sense amplifiers and circuits for accurate reference voltage generation.

2.1.1.2 Gillingham's MLDRAM — Sequential Sensing with Local Referencing using Switch Matrix Method

Gillingham proposed an MLDRAM scheme that uses sequential sensing with local reference signal generation [14]. Data is sensed one bit at a time starting with the most significant bit (MSB). The result from sensing the MSB is used to generate the reference signal for sensing the least significant bit (LSB). Thus, a two-step sequential sensing and local generation of reference levels method is used in Gillingham's

[7] Furuyama's MLDRAM claims to have an access time of 170 ns under typical operating conditions [13]

MLDRAM. Figure 2.6 shows a bitline configuration from Gillingham's MLDRAM circuit.

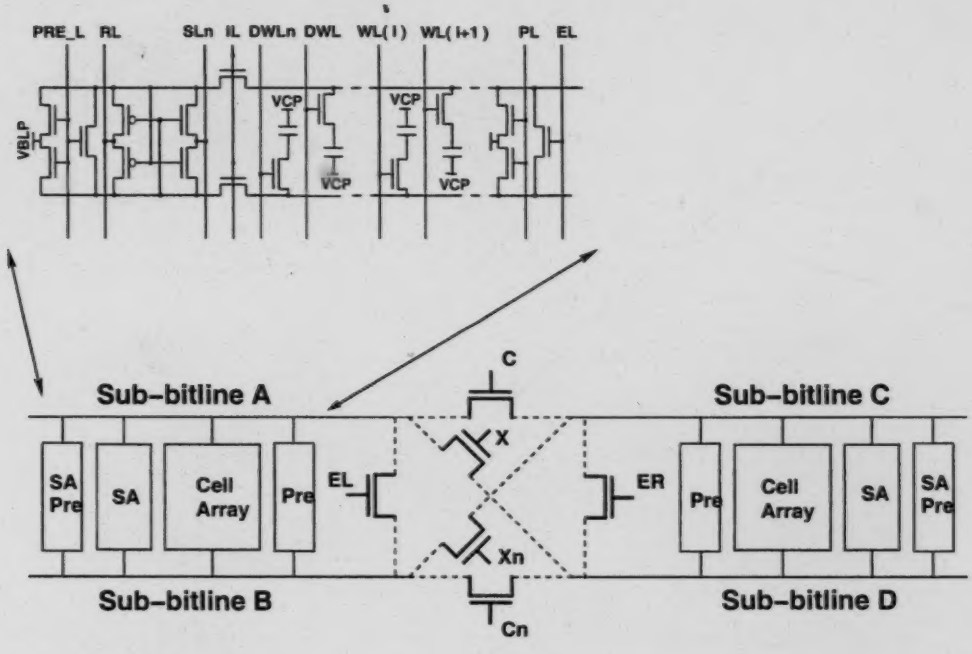


Figure 2.6: Schematic of Gillingham's MLDRAM showing four sub-bitlines [14]

Four sub-bitlines (A, B, C, D) are shown in Figure 2.6. The four sub-bitlines can be connected using the C, Cn, X, Xn, EL and ER signals to the corresponding switches. The switches make up a switch matrix that will be used for reference generation for sensing and restoring the MSB and the LSB. The sub-bitlines pairs on the left and right of the switches are identical in size and capacitance. Each block contains a bitline precharge circuit, a sense amplifier, dummy cells and memory cells. The dummy cells in Gillingham's MLDRAM design are used for balancing the sub-bitline capacitances and capacitively injected noise signals during the sense and restore operation [14], unlike in Furuyama's MLDRAM design where the dummy cells are used to store the reference values [13]. The sense amplifiers at the end of each sub-bitline block can be connected or disconnected via the signals IL and IR to the isolation transistors.

Gillingham's MLDRAM does not need the thermometer code data conversion

as shown in Figure 2.2 for Furuyama's MLDRAM explained in section 2.1.1.1. For Gillingham's four-level MLDRAM, the two bits are sensed sequentially, with the MSB compared against $\frac{1}{2}V_{DD}$, and the LSB compared against one of two possible reference levels ($\frac{5}{6}V_{DD}$ or $\frac{1}{6}V_{DD}$) created from the full value (V_{SS} or V_{DD}) of the sensed MSB. Figure 2.7 shows the reference values and data voltage values for storage of two bits of data per cell using Gillingham's method.

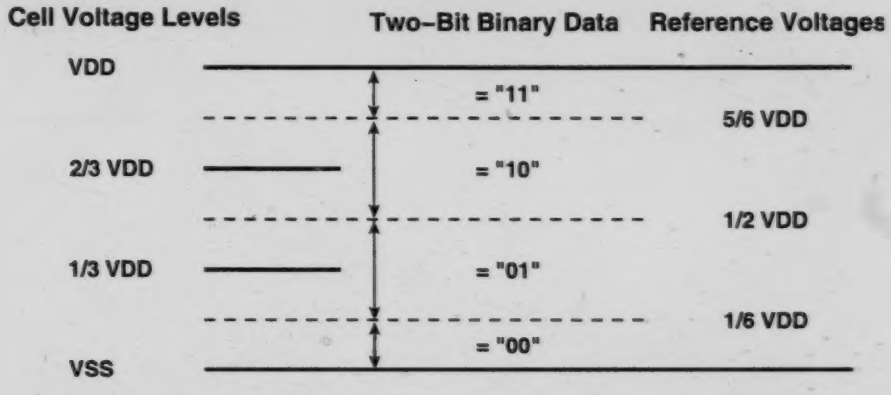


Figure 2.7: Reference and data voltage levels for Gillingham's MLDRAM

The following events make up the sensing operation for the MSB, which are also illustrated in Figure 2.8:

(a) Assuming that the memory cell addressable from WLi contains one of the four nominal data voltage values, the bitlines are first isolated from one another and the sense amplifiers and then precharged to $\frac{1}{2}V_{DD}$. The $\frac{1}{2}V_{DD}$ precharge value is used as the reference level for MSB sensing.

(b) After the precharge stage, the addressed data cell wordline can then be asserted after the corresponding dummy wordline is deasserted. The dummy wordline transition is used to cancel out charge injection via the cell access transistor capacitance to the sub-bitlines when the data wordline is asserted or deasserted. Having dummy cells also ensure that the number of cells connected to each sub-bitline at any one time is the same, thus balancing the sub-bitlines capacitively. Note that the dummy wordlines on the other three sub-bitlines are asserted at

all times. Other common mode signals, such as the bitline equalize and sense amplifier isolation signals, do not unbalance the bitlines due to the folded bitline architecture.

After the addressed wordline is asserted, the Cn switch is pulsed on then off so that the bitline adjacent to the bitline containing the addressed cell also shares the same diluted data voltage. The data value is contained in the adjacent bitline for use later on in the reference generation for the LSB sensing.

(c) Referring to Figure 2.8, sub-bitlines B (and D) now contain the diluted voltage value from the addressed cell while sub-bitline A retains the precharge value of $\frac{1}{2}V_{DD}$. To complete the sensing of the MSB, the sense amplifier-bitline isolation switches are asserted to connect the left sense amplifier to the sub-bitlines so that the voltage value on sub-bitline B can be amplified against the precharge value on sub-bitline A.

(d) The addressed wordline is then deasserted to store the full value of the MSB on the original memory cell on sub-bitline B. The dummy wordline on the same sub-bitline is then asserted to balance out the charge injection from deasserting the addressed wordline. The memory cell in sub-bitline B now contains the full value (V_{SS} or V_{DD}) of the MSB. The left sub-bitlines are then isolated and precharged to $\frac{1}{2}V_{DD}$ while sub-bitline C is still untouched at $\frac{1}{2}V_{DD}$ and sub-bitline D contains the original diluted cell value before sensing.

Now the four sub-bitlines are ready for the reference generation and sensing of the LSB. The LSB reference level can be obtained by dumping the full sensed MSB value onto three sub-bitlines. The steps shown in Figure 2.9 are explained in the following:

(a) Starting at the point where the MSB value is sensed, sub-bitlines A and C are precharged to $\frac{1}{2}V_{DD}$. The EL switch is left asserted as in during the precharge stage, so that sub-bitlines A and B are still connected. Next, the dummy wordline on sub-bitline B is deactivated shortly before the wordline, WLi, on the

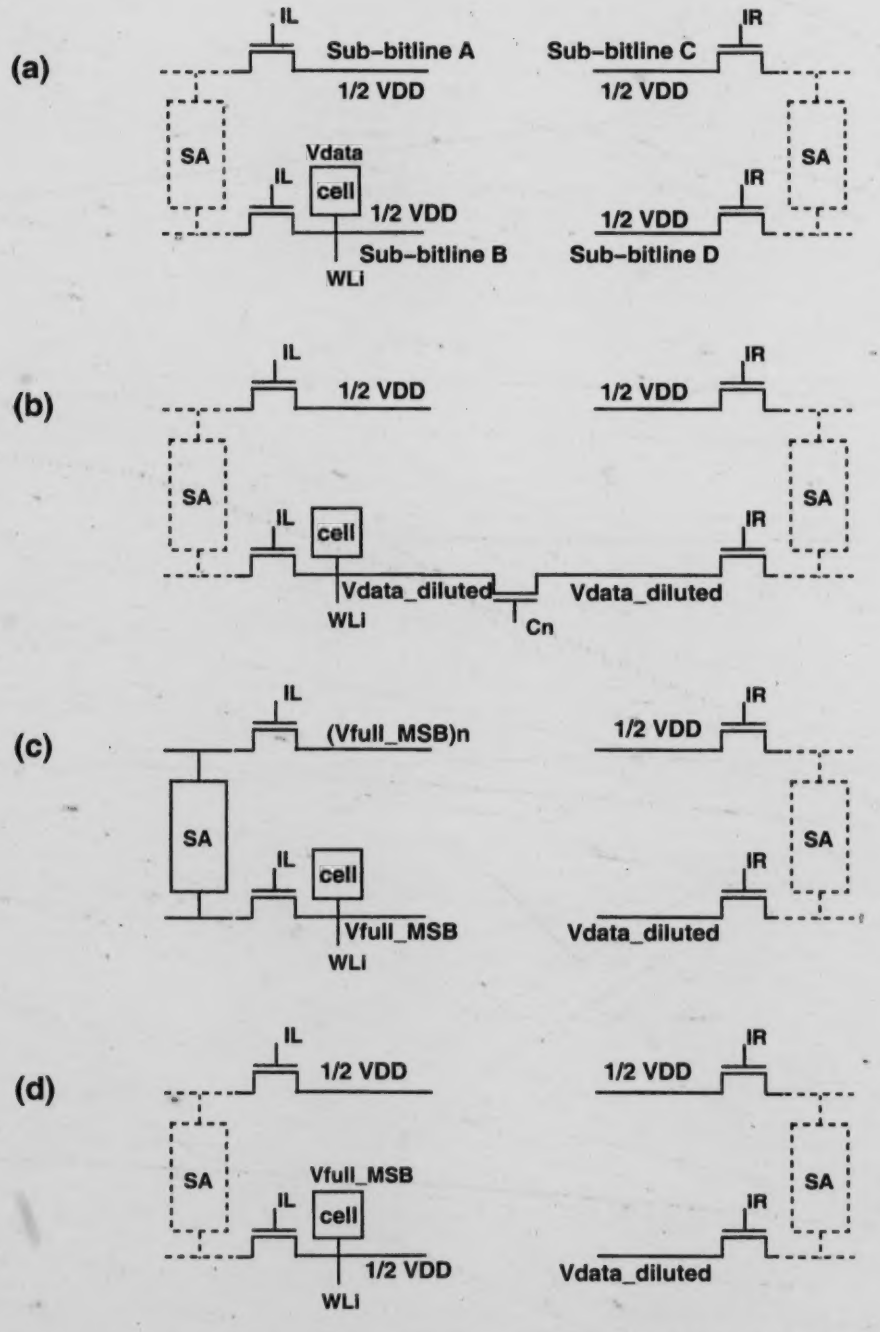


Figure 2.8: MSB sensing steps in Gillingham's MLDRAM

same sub-bitline is asserted. At the same time, transistor switch C is asserted so that sub-bitline C is connected to both sub-bitlines A and B. The memory cell value (V_{SS} or V_{DD}) is now diluted across three sub-bitlines. This charge sharing scheme is able to produce the appropriate reference level depicted in Figure 2.7, by Equation (2.1) on the resultant voltage on the bitline after charge sharing, to give the following Equation (2.3).

$$V_{REF} = \left(S - \frac{V_{DD}}{2}\right)\left(\frac{C_{cell}}{3C_{bitline}}\right) + \frac{V_{DD}}{2} \qquad (2.3)$$

where

$$S = V_{DD} \quad \text{or} \quad V_{SS}$$

In Equation (2.3), $\frac{V_{DD}}{2}$ is the bitline precharge voltage, C_{cell} is the memory cell capacitance and $C_{bitline}$ is the capacitance of a sub-bitline including a memory cell.

(b) With the reference for LSB now generated and sub-bitline D still containing the original voltage value of the addressed memory cell, the right sense amplifier is then used to sense the original cell voltage against the LSB reference voltage. This is done by isolating all sub-bitlines and connecting only the right sense amplifier to sub-bitlines C and D through the IR switches. The full value of the LSB is now contained in sub-bitline D.

(c) Next, the full MSB value from the left sense amplifier can be put onto sub-bitline B by connecting the sense amplifier through the IL switches. At this point, the two-bit binary data depicted in Figure 2.7 are available on sub-bitlines B (MSB) and sub-bitlines D (LSB) to be read out to the databus. The right sense amplifier can be deactivated after the sensing of the LSB is done.

The restore or write operation in the Gillingham scheme is similar to that of Furuyama's MLDRAM explained in section 2.1.1.1 in that charge sharing is used to recreate the original data value from the fully sensed data bits. The following

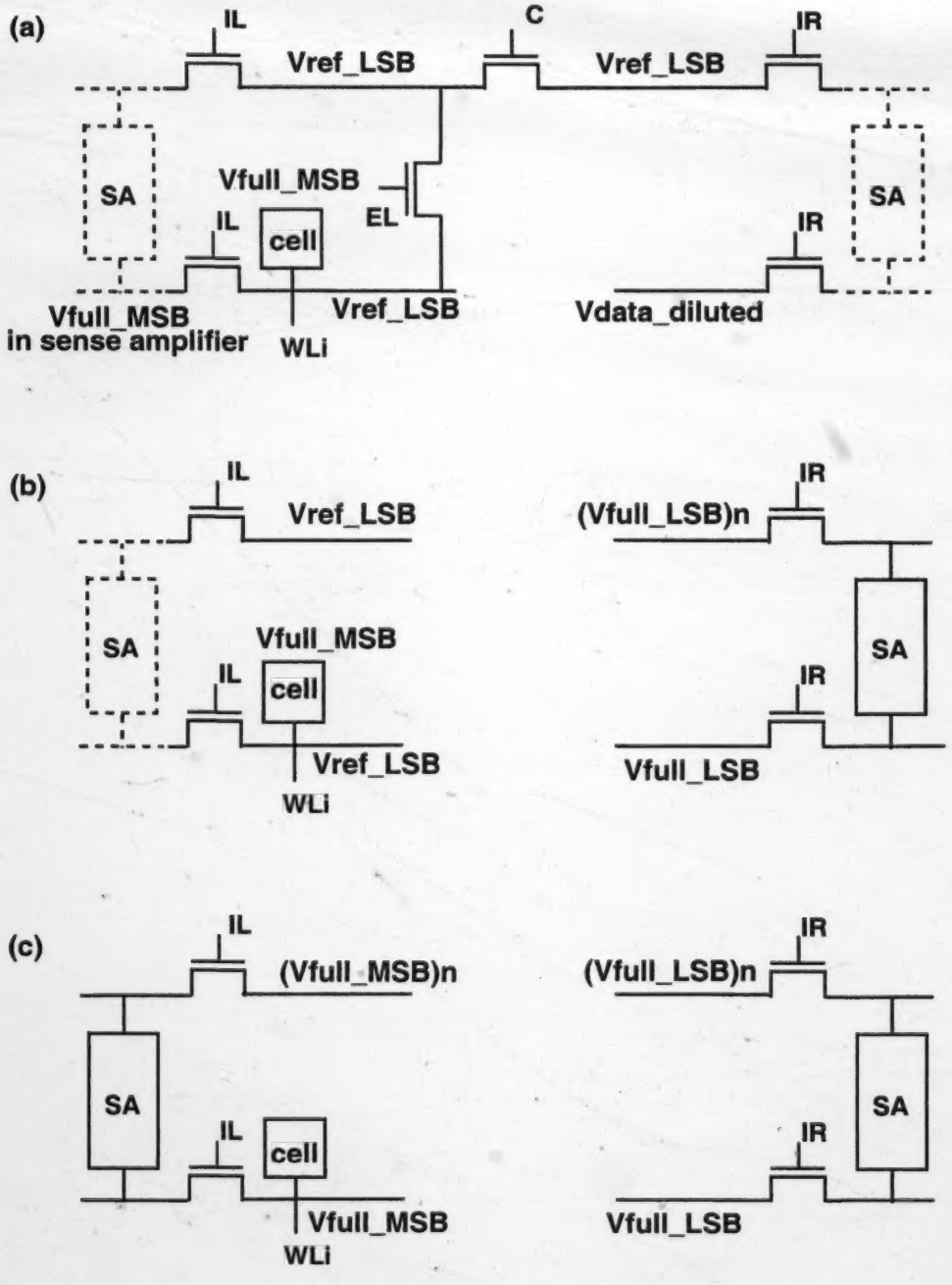


Figure 2.9: LSB sensing steps in Gillingham's MLDRAM

explains how the restore operation can be done and Figure 2.10 illustrates the operation:

(a) When the full MSB and LSB values are available on sub-bitlines B and D, respectively, the original data value can be restored into the memory cell by charge sharing operations on the sub-bitlines. At this point, sub-bitline B would contain the full MSB value while sub-bitline D holds the full LSB value from an earlier sensing. To begin the restore operation, sub-bitlines B and C are first reconnected using the Xn signal. With this connection, the left sense amplifier will be able to charge both sub-bitlines B and C to the full MSB value. Sub-bitline D is still isolated and holds the full LSB value.

(b) Then charge sharing occurs between two sub-bitlines with the MSB value and one sub-bitline with the LSB value when the ER switch is asserted. The resultant voltage from this charge sharing operation is the original data value that will be rewritten into the memory cell when the addressed wordline is de-asserted.

(c) After the restore operation, the four sub-bitlines can be precharged to $\frac{1}{2}V_{DD}$ in preparation for the next sensing cycle.

Gillingham's MLDRAM [14] design uses local reference signal generation circuitry in the memory array, unlike the global generated references in the Furuyama design. Degradation of signal quality should be less severe. The locally generated reference voltage also makes the circuits less susceptible to errors due to process skews since the reference signals are generated from circuits on the same semiconductor area as the sense circuitry.

However, the sequential sensing method proposed with the design requires more steps. Also, significant area is taken up by the required switch matrix although the MLDRAM scheme requires two sense amplifiers instead of the three in Furuyama's design [13]. The switching logic is more complex than in the Furuyama design and the additional switching operations required for bitline balancing could introduce more noise to the already noisy DRAM environment.

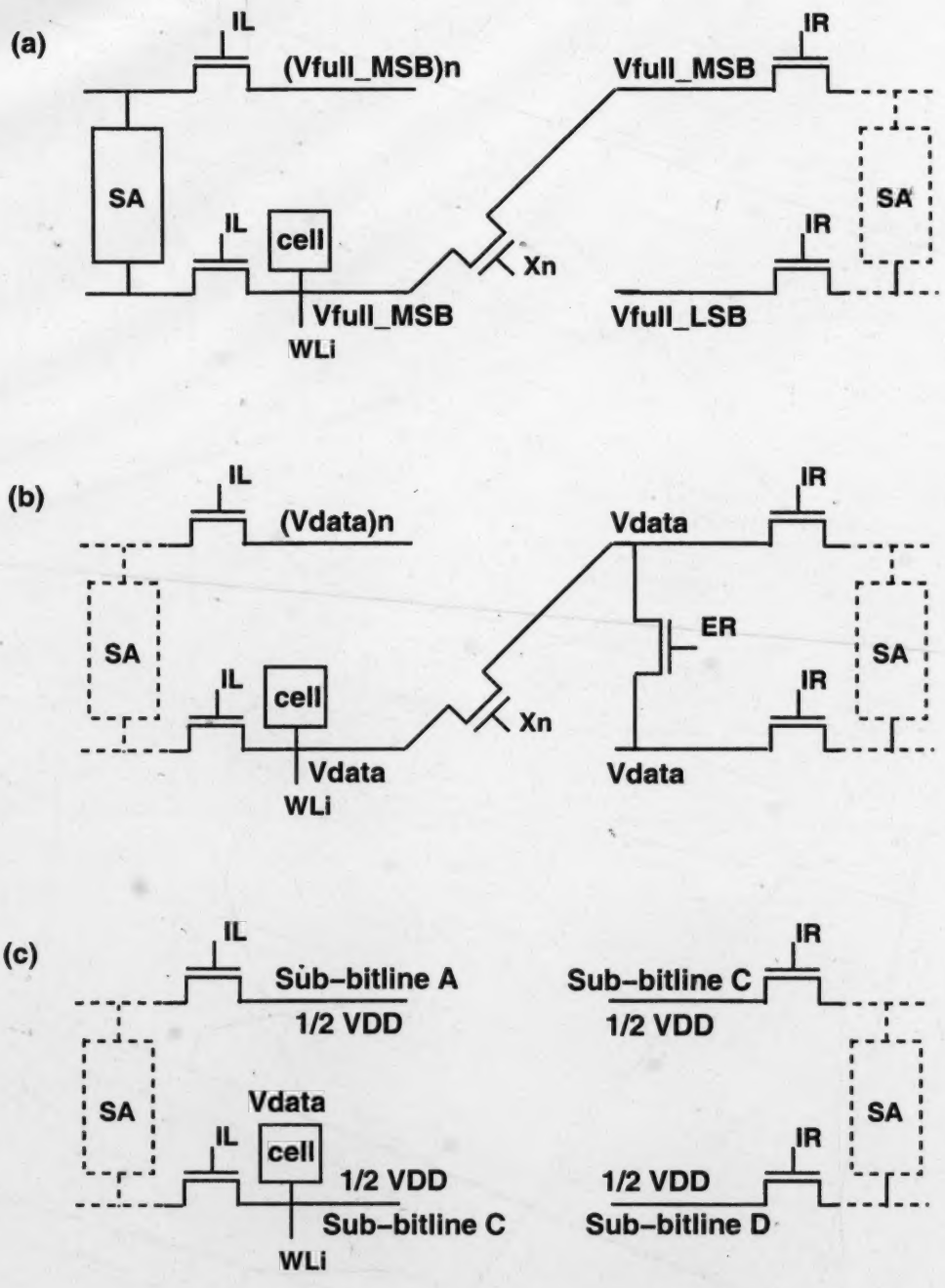


Figure 2.10: Data restore steps in Gillingham's MLDRAM

2.1.1.3 Okuda's MLDRAM — Sequential Sensing with Local Referencing using Capacitive Coupling Method

An MLDRAM technology developed by Okuda et al. from NEC uses sequential sensing with local reference generation that involves capacitive coupling of bitlines [23]. This technology is used in one of NEC's prototype 4 Gbit DRAM designs published in [23].

Okuda's MLDRAM uses the same data conversion thermometer coding as Gillingham's MLDRAM explained in section 2.1.1.2. Figure 2.7 from section 2.1.1.2 shows the reference and data voltage levels. Okuda's MLDRAM uses a similar sensing and restoring scheme as Gillingham's MLDRAM. The sensing of the MSB and LSB data is sequential, as in [14], except for the usage of charge-coupling capacitors instead of local bitline capacitance ratio.

In the charge-coupling method, cross-coupled capacitors are used to provide enough capacitance to the sub-bitlines so that the correct LSB reference value can be generated from the value of the MSB. The sub-bitline capacitances are weighted 2:1, with the sub-bitlines generating the sensed MSB value being the more heavily weighted one. Note that Gillingham's MLDRAM design also uses this ratio on the sub-bitlines for reference generation and restore operations.

Figure 2.11 shows the sub-bitlines in a memory array column in Okuda's MLDRAM design. The left and right sub-bitline pairs can be separated or connected by transfer switches while the charge-coupling capacitors connect them. Similar to Gillingham's MLDRAM design, the left and right sub-bitline blocks each have precharge and sense amplifier circuits, and a memory array. Okuda's MLDRAM design does not make use of dummy cells for charge balancing.

Figure 2.11 can be used to explain the sense and restore schemes. The sensing scheme uses charge-coupling to create the appropriate voltages on the bitlines. The sensing operation starts with all bitlines precharged to $\frac{1}{2}V_{DD}$ and the transfer switches asserted. Next, the data voltage of from the addressed cell is distributed to the bitlines when the appropriate wordline is asserted. Note that the sub-bitlines B and C, and sub-bitlines A and D are connected by coupling capacitors C_C. At

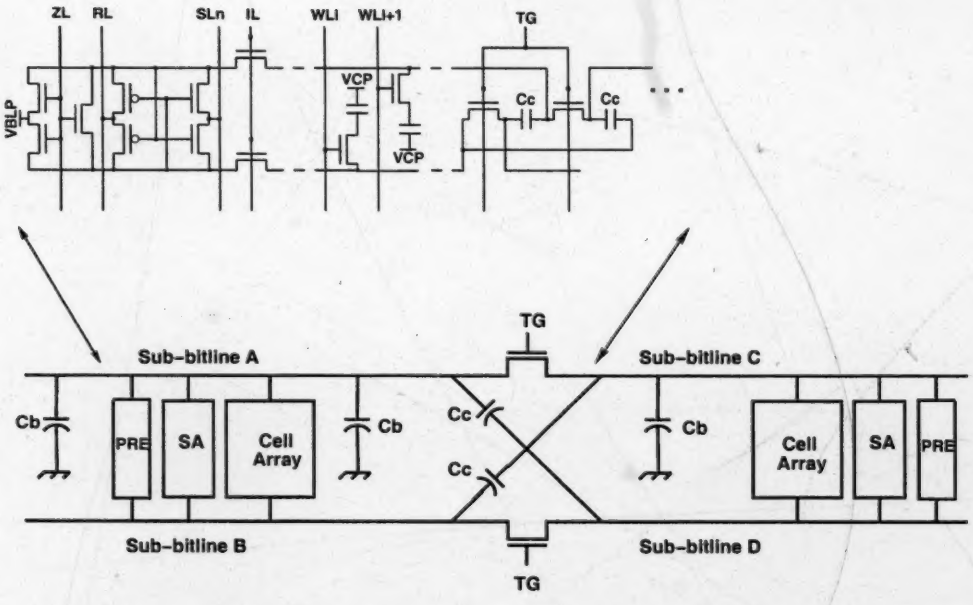


Figure 2.11: Schematic of Okuda's MLDRAM showing sub-bitline pairs [23]

this point, the transfer switches are turned off, isolating the left and right sub-bitline pairs. The left sense amplifier is then turned on so that the MSB value can be sensed against the $\frac{1}{2}V_{DD}$ reference value on sub-bitlines A.

Depending on the full MSB value sensed, one of sub-bitlines C or D, will experience a $\frac{1}{2}V_{DD}$ change and the other, an equal and opposite change, through the coupling capacitor connections. This is due to the in-built 2:1 bitline capacitance ratio. And, with the coupling capacitors in place, the bitline voltages on sub-bitlines C and D will also change accordingly so that the correct LSB value will be sensed when the right sense amplifier is activated.

To restore the data value into the memory cell, the sense amplifiers are turned off and the sub-bitlines are reconnected by asserting the transfer switches, so that charge sharing can occur to create the correct data value. The addressed wordline can then be deasserted so that the value is stored into the memory cell.

To determine the value of the charge-coupling capacitor, we first have to understand that the voltage on the sub-bitlines are proportional to the bitline capacitance and storage capacitance. Realizing also that the absolute signal level of data volt-

ages are $\frac{1}{3}V_{DD}$ apart, it can be deduced from Equation (2.3) of the reference value generation and Equation (2.4) of data voltage restoration taken from [14] that the charge-coupling capacitors are to be made $\frac{1}{3}$ the value of the storage capacitance [23].

$$V_{RESTORE} = 2\frac{S}{3} + \frac{M}{3} \tag{2.4}$$

where

$$S = V_{DD} \text{ or } V_{SS}$$

$$M = V_{DD} \text{ or } V_{SS}$$

In Okuda's MLDRAM, a hierarchical bitline architecture is employed where the 2:1 bitline ratio is obtained by connecting segments of bitlines together [23]. Figure 2.12 shows the organization of the bitline-connect switches and the time-shared sense amplifiers. The TGA and TGB transfer switches separate or connect the sub-bitline segments to give the 2:1 ratio on the bitlines for sensing. At the edge of the sub-bitline arrays there are sense amplifier arrays which operate in a time-shared sensing scheme. The sensing scheme is time-shared in that any one of every four sets of bitline pairs are connected to the sense amplifier at any one time during sensing. Control switches within the sense amplifier circuitry transfer data voltages from the main bitlines to the sub-bitlines, and vice versa. The sense amplifiers used in this scheme are offset-cancelled sense amplifiers [12, 23].

As in Gillingham's sequential sensing MLDRAM, Okuda's MLDRAM will tend to be slower than a parallel sensing MLDRAM. Also, Okuda's MLDRAM is more sensitive to process variations since it will likely be difficult to get accurate bitline capacitance to automatically generate accurate voltages for LSB sensing. Coupling capacitance variations will not be correlated with variations in the sub-bitline capacitance. In spite of the obvious simplicity of Okuda's MLDRAM design and sensing scheme, the design may not be robust against process variations and reduction in noise margins due to imbalances in bitline capacitances incurred by the capacitive-coupling method [6].

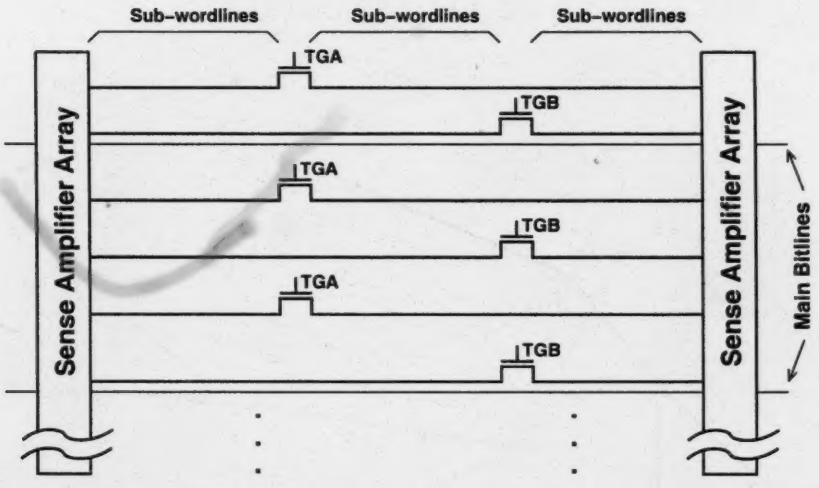


Figure 2.12: Okuda's MLDRAM bitline hierarchy and time-sharing sense amplifier architecture [23]

2.2 Multilevel DRAM Chips from the University of Alberta

At the University of Alberta, four previous MLDRAM chips have been designed based on the experiences of the published MLDRAMs described earlier. Gershom Birk's two MLDRAM test chips (ML1 and ML2) were based on Gillingham's MLDRAM and Furuyama's MLDRAM. Albert Chan's test chip (ML3) implemented a four-level version of Birk's proposed parallel sensing MLDRAM. Yunan Xiang's MLDRAM (ML5[8]) is an improved version of Birk's design, with variable cell capacity feature, expandable from two levels to three, four and six levels.

2.2.1 Birk's Four-level MLDRAM

In his thesis research work, Birk [5] studied three MLDRAM schemes and architectures available in the literature and propose a new MLDRAM scheme that is more robust and faster, compared to the alternative MLDRAMs. The three MLDRAM schemes investigated and described in his thesis were Furuyama's [13], Gillingham's [14] and Okuda's [23]. Aoki's [1] MLDRAM scheme was not taken

[8]ML4 was never sent for fabrication.

into account in the evaluation since the proposed staircase method requires lengthy sensing time and lacks the noise immunity dictated by multilevel sensing.

Birk's MLDRAM scheme is based upon the advantages from Furuyama's and Gillingham's MLDRAM. It is a combination of the fast parallel sensing of Furuyama's design and the local reference generation ideas from Gillingham's MLDRAM. The new scheme implements fast single-step flash-conversion sensing (similar to the parallel sensing in Furuyama's design [13]) and local reference generation via charge-sharing techniques between adjacent bitlines as was used by Gillingham in his MLDRAM design in [14].

The bitlines in Birk's design are divided into three equal segments as in Furuyama's design [13]. Each segment is identical capacitively and are connected by transfer switches. There are two sub-bitline configurations that used in the design. Figure 2.13 shows the two sub-bitline types. One contains the "reference" wordlines while the other contains the "generate" wordlines. The "reference" and "generate" type sub-bitlines differ by their connections to the reference and generation wordlines. In the "reference" type sub-bitlines, the gate connections to the "generate" wordlines are missing. The gate connections to the "reference" wordlines are in turn missing for the "generate" type sub-bitlines. Figure 2.14 shows how the two types of sub-bitlines are put together to form a 3-by-3 array. The middle sub-bitlines (M) are of the "reference" type and are used to store reference voltages. The top and bottom sub-bitlines are the "generate" type. They are used for charge-balancing and noise-injection cancellation during sensing. The reference and generate type cells are similar in all respect to a memory cell in the array. The sub-bitlines are also labeled as shown in Figure 2.14. The nine sub-bitline pairs can be connected via transfer switches.

In between the L, C and R groups of sub-bitline pairs, there are additional switches that are connected to different voltage values. These switches are used to precharge parts of the sub-bitlines with the voltages (VDC) connected to these switches as shown in Figures 2.13 and 2.14. After this mini precharge operation, the bitlines with the different values are connected together so that reference voltages

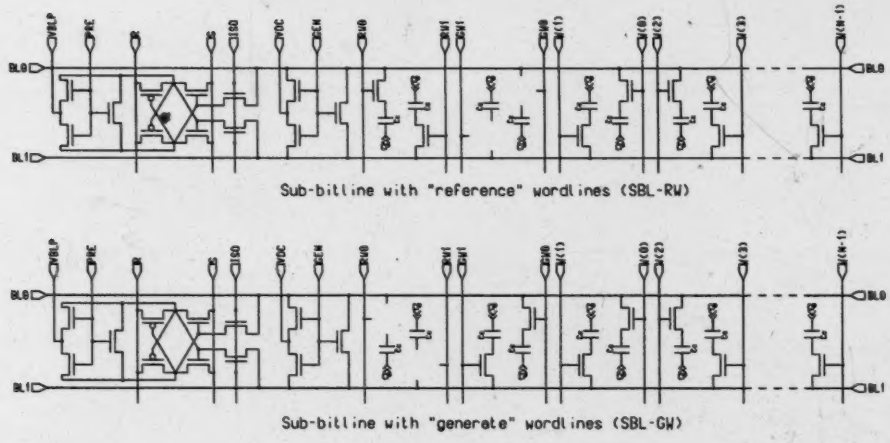


Figure 2.13: "Reference" and "generate" sub-bitline types for Birk's MLDRAM design [5]

can be formed through charge-sharing between the sub-bitlines. The charge-sharing gives the L group of sub-bitline the reference value of $\frac{1}{6}V_{DD}$, the C group of sub-bitlines the reference value of $\frac{1}{2}V_{DD}$ and the R group of sub-bitlines the reference value of $\frac{5}{6}V_{DD}$. The reference voltages are stored in the middle sub-bitlines (group M) of the L, C and R groups of sub-bitlines. The group M sub-bitlines are of the "reference" type sub-bitlines. This charge-sharing local reference voltage generation idea is reminiscence of Gillingham's method in [14].

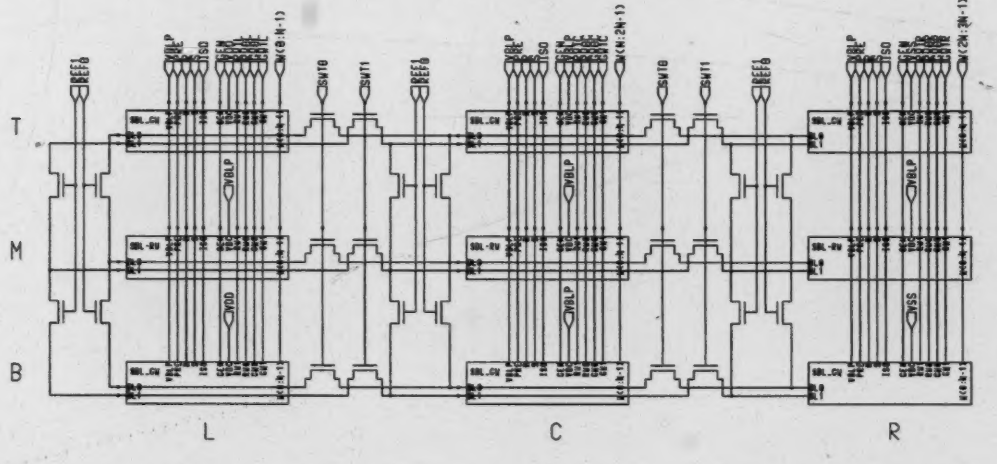


Figure 2.14: Schematic of Birk's MLDRAM scheme showing nine sub-bitline pairs [5]

Birk's MLDRAM design stresses on capacitive balance and charge-injection cancellation in the bitlines during sense and restore. To achieve these goals, the generate cells are carefully and timely asserted so that each sub-bitline have the same capacitance equal to the parasitic capacitance of the bitline and one memory cell, at any one time during sense, restore and reference generation operations.

Reference Generation Before the sense and restore operations, the reference voltages have to be generated. To prepare for these operations, the bitlines and sense amplifiers are first precharged to $\frac{1}{2}V_{DD}$. At this time, the sub-bitlines are shorted together horizontally and vertically through the transfer switches. After the bitline precharge operation, all sub-bitlines are isolated and all reference and generate wordlines are asserted so that all sub-bitlines have equal parasitic capacitances. Then the sub-bitlines can be precharged to their respectively set voltages for charge-sharing for reference voltage generation. Figure 2.15 shows the state of the sub-bitline connections for reference generation before charge-sharing. When the mini precharge on the sub-bitlines are completed, the voltages are charge-shared along the L, C and R groups of sub-bitlines by asserting the vertical switches. The charge-sharing operation creates the appropriate reference voltages shown in Figures 2.2 and 2.7. These reference voltage values are also used in Furuyama's [13] and Gillingham's [14] MLDRAMs. After charge-sharing in the vertical direction along the L, C and R groups of sub-bitlines, the reference wordlines are deasserted so that the reference voltages can be stored. In Birk's design, the voltage $\frac{1}{6}V_{DD}$ (from a combination of V_{SS}, $\frac{1}{2}V_{DD}$ and V_{SS}) is created in the L sub-bitlines, $\frac{1}{2}V_{DD}$ (from a combination of three $\frac{1}{2}V_{DD}$ voltages) in the C sub-bitlines and $\frac{5}{6}V_{DD}$ (from a combination of V_{DD}, $\frac{1}{2}V_{DD}$ and V_{DD}) in the R sub-bitlines.

Sensing To prepare for sensing, all sub-bitlines are first precharged to $\frac{1}{2}V_{DD}$. During sensing, say when an even wordline is addressed, all complementary sub-bitlines are disconnected in the horizontal direction using the appro-

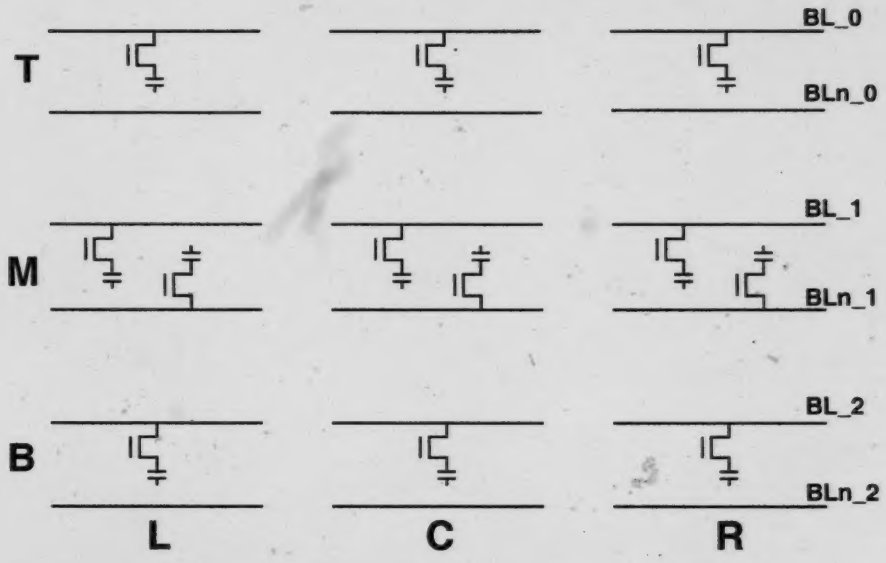


Figure 2.15: Birk's sub-bitline connections and reference-generate wordline connections for the reference generation operation [8, 42]

priate transfer switches. The true sub-bitlines, however, are left still connected horizontally. At the same time, to ensure that all the sub-bitlines are experiencing the same parasitic capacitance as the sub-bitlines connected to the addressed wordline, all the complementary sub-bitlines are connected together vertically via the appropriate transfer switches. Figure 2.16 shows the bitline configuration in the case where wordline WL0, residing in sub-bitline TL, is addressed. Just before sensing, the contents of the memory cell, addressed at WL0 in our example, is dumped onto the three true bitlines (true bitline since WL0 is an even wordline) and charge-shared horizontally. At the same time, with the configuration described above, the reference voltages from the complement reference bitline of row M, are dissipated vertically along the L, C and R sub-bitline groups. The actual sensing occurs when all sub-bitlines are isolated. The three thermometer bits can then be sensed in parallel by activating the sense amplifiers along section T. The sensing of the cell voltage from the memory cell at WL0, is done against the three reference voltages available in

the complementary sub-bitlines of TL, TC and TR.

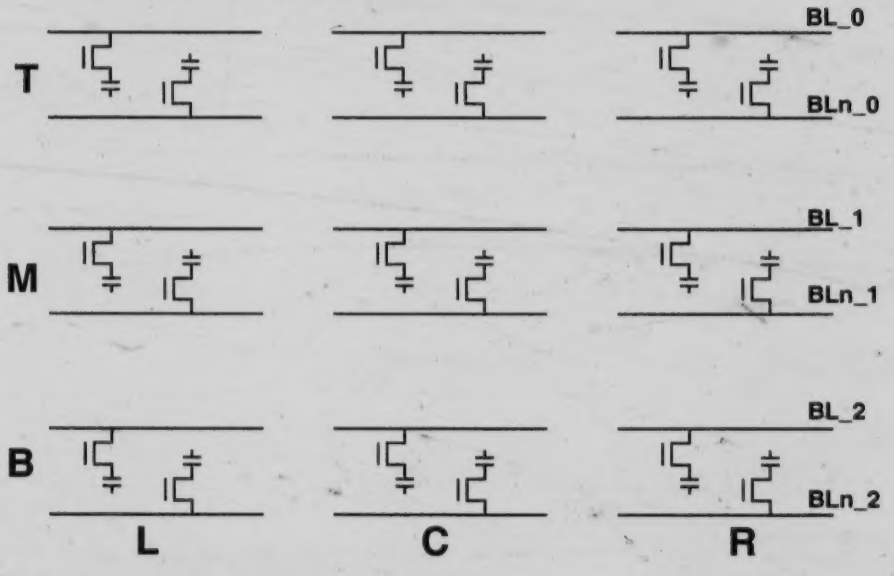


Figure 2.16: Birk's sub-bitline connections and reference-generate wordline connections for the sensing operation [8, 42]

Restore Still assuming that the addressed wordline is WL0, Figure 2.17 illustrates the sub-bitline configuration in preparation for restore. The addressed wordline is kept asserted and all true sub-bitlines are reconnected horizontally after the sensing operation. Reconnecting the true sub-bitlines horizontally after sensing will charge-share the sensed voltages on the sub-bitlines to create one of the four analog data levels on the bitline addressed by the wordline. Notice in Figure 2.17 that all three full (three sub-bitlines connected together) and true bitlines are each connected to three memory cells by asserting the appropriate reference and generate wordlines. This is so that all the full bitlines are capacitively equal. To complete the restore operation, the addressed wordline is deasserting so that the analog data voltage is restored into the memory cell.

Birk's MLDRAM design has the advantage, to a first order approximation, of charge injection immunity. His PSPICE simulations have shown that the switch

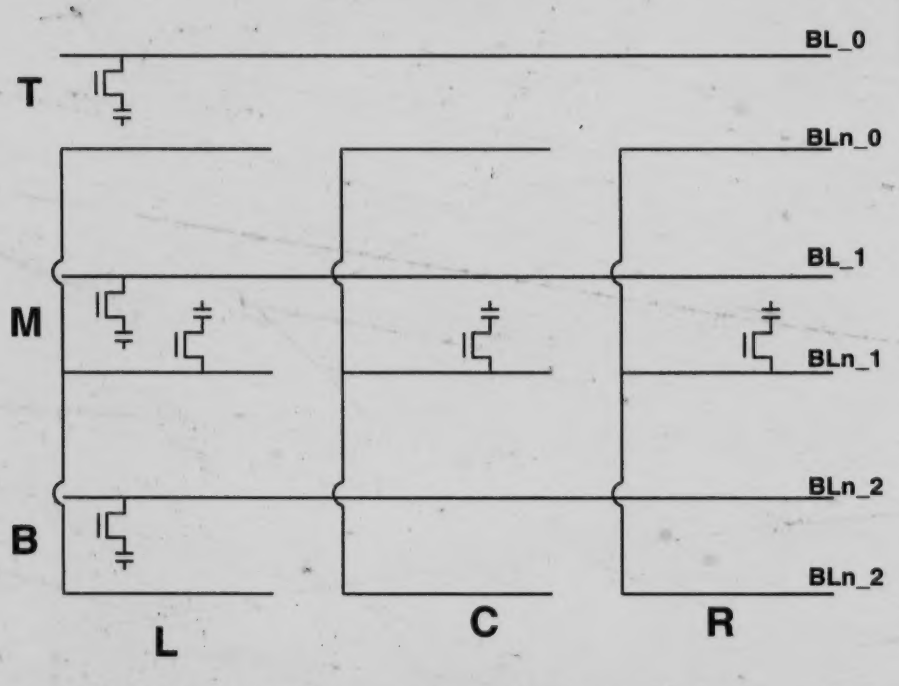


Figure 2.17: Birk's sub-bitline connections and reference-generate wordline connections before the completion of the restore operation [8, 42]

activations that cause unequal charge injections into the bitlines are balanced by the activations of the other switches such as the activation of the "reference" and "generate" wordlines. The design uses local reference voltage generation and fast parallel sensing. Nevertheless, the area overhead is large due to the required switch matrix and reference generation circuits, and may render it uncompetitive to the DRAM market. The area disadvantage, however, brings forth the need for further research and building of test chips for characterization of the circuits. The need has formed the basis of MLDRAM research at the University of Alberta, which has resulted in two working test chips [8, 43]. The test chips will be explained next in the following sections.

2.2.2 Chan's Implementation of Birk's Four-level MLDRAM

Birk's MLDRAM design [6] is realized in ML3, Chan's implementation of the test chip [8]. ML3 uses a parallel sensing scheme, a switch matrix and charge sharing

techniques for reference generation as was designed by Birk. Birk's MLDRAM scheme is expandable to more than 2 bits per cell but only four-level operation is implemented in the chip.

The design uses two different sub-bitline types, called reference and generate sub-bitlines, for storing the reference voltages, charge balancing and charge injection cancellation. The reference and generate sub-bitline types are arranged in a 3-by-3 array with a switch matrix to connect the bitlines vertically and horizontally. A simplified diagram of the chip floorplan is shown in Figure 2.18 [8].

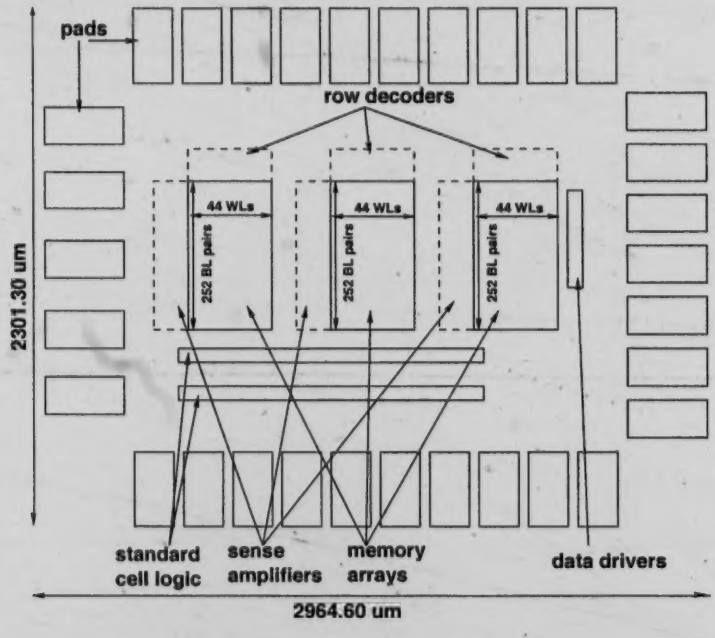


Figure 2.18: Chan's ML3 test chip floorplan [8]

In Chan's test chip, there are 132 wordlines, 12 reference and generate wordlines and 252 bitline pairs. The wordlines and bitlines are distributed in equally numbers, horizontally and vertically in the sections making the 3-by-3 array. Eight databus pairs are used to access the 32 column-select lines that used to select from the 252 bitlines. A block diagram of the connections between the constituent components in the test chip is illustrated in Figure 2.19 [8]. The chip is generally divided into the core and the periphery. The core contains the memory cell array, sense amplifiers, row decoders and wordline drivers; while the periphery contains address

(X and Y) decoders, databus decoders, input/output (IO) buffers, and reference-generate wordline decoders.

2.2.2.1 ML3's Core

The core floorplan is shown in Figure 2.20 [8]. In the figure, the bitlines run horizontally over the sense amplifiers. This is necessary due to the architecture and layout of the design. The design of the core also includes friendly cells, and folded and twisted bitlines architecture. Also shown in Figure 2.20 is the bitline twists employed in the design to reduce capacitive coupling noise.

The wordline drivers, including the reference and generate wordline drivers are basically bootstrap drivers as shown in Figure 2.21. A novel multiplexing and decoding technique has been used here to reduce the number of external pins required for controlling the reference and generate wordlines in each of the sections in the memory array. Since there are only three and two distinct types of waveforms needed to control the reference and generate wordlines, respectively, a reduced number of pins can be used to generate these waveforms automatically. Additionally, the two distinct waveforms going to the generate wordlines are the same as the two out of three waveforms going into the reference wordlines [8]. Therefore, the signals coming out of three wordline drivers can be multiplexed to create one of the three essentially distinct waveforms to each of the reference and generate wordlines. The three external input sources are coined the RGX signals — RGX1, RGX2 and RGX3 — where R stands for reference, G stands for generate and X represents wordlines going in the X direction. There are only a few distinct waveforms since we know how the reference and generate wordlines should behave during the precharge, sense and restore operations. The RGX signals are explained in more detail in Appendix F.

Since the bitlines are twisted, row and column address scrambling has to be carefully ironed out. Chan used a simple scrambling function for the row and column addressing to ease vector generation [8].

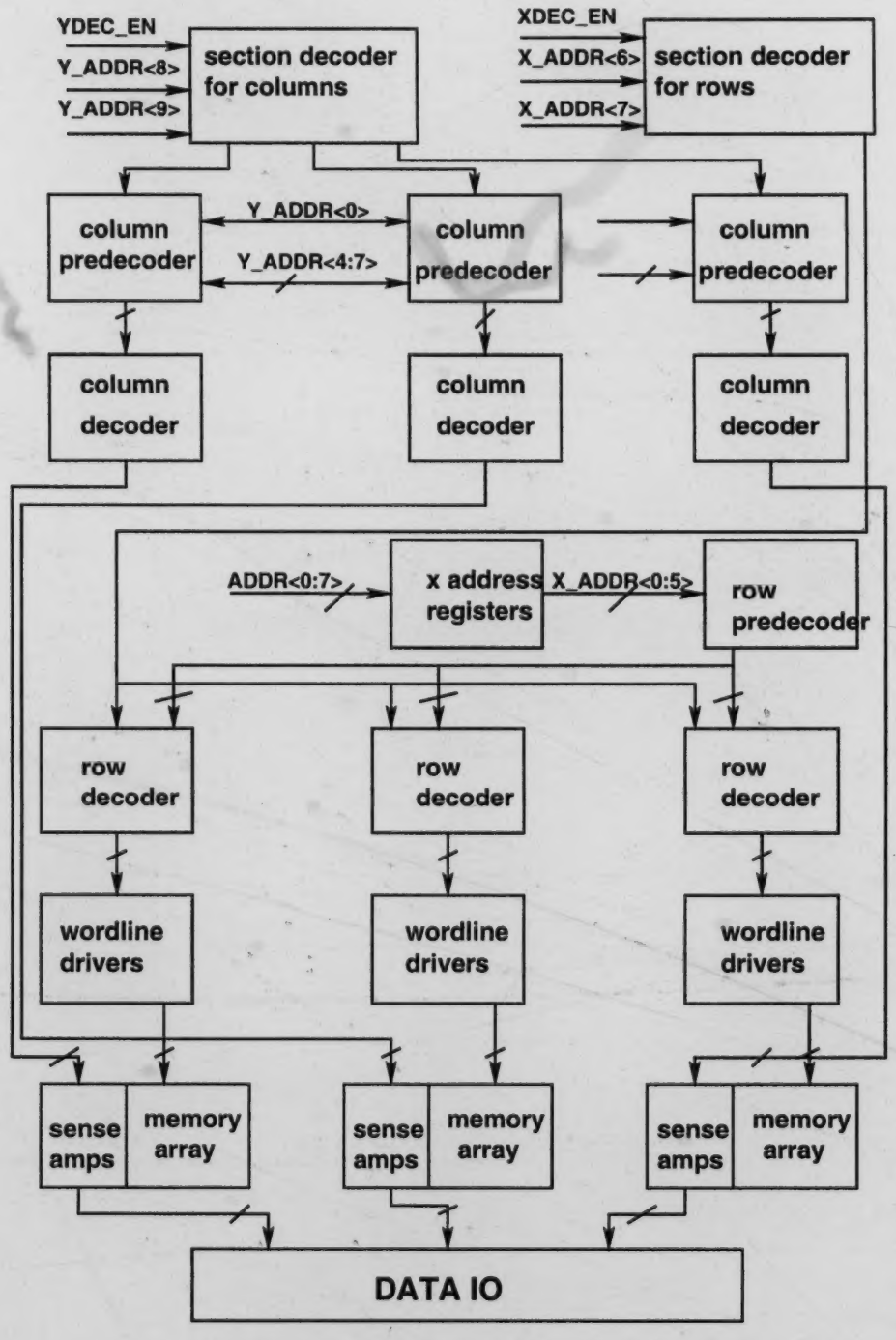


Figure 2.19: Simplified block diagram of ML3 [8]

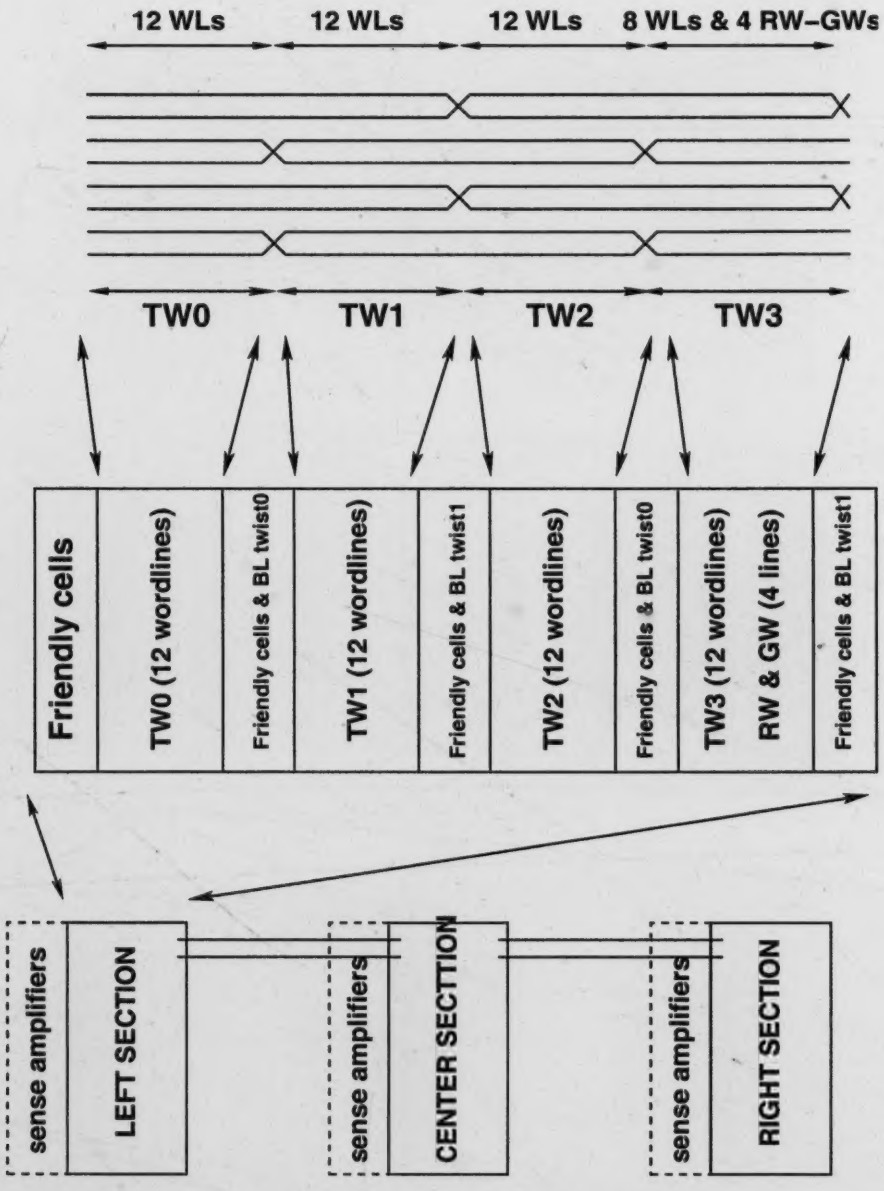


Figure 2.20: Chan's ML3 memory core floorplan showing sub-bitline connections between sections [8]

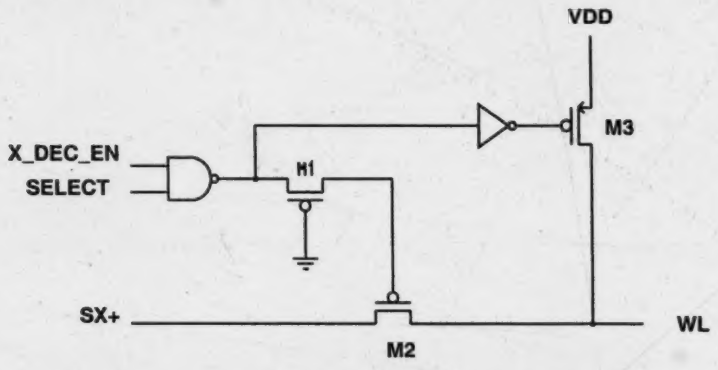


Figure 2.21: Wordline boost driver used in ML3 [8]

2.2.2.2 ML3's Periphery

ML3's periphery mainly consists of address and databus decoders made out of standard cells available from Canadian Microelectronics Corporation (CMC) available in Taiwan Semiconductor Manufacturing Corporation's (TSMC's) 0.35-μm process technology. The column access in ML3 are made from eight address bits. Eight address bits are needed since there are 252 bitlines in ML3. To determine which sections are addressed, two additional address bits are used. The sections are determined first before the respective bitlines are addressed.

A databus is available to two sense amplifiers, selectable by two column-select lines. There are eight databus lines available to connect to any two column-select lines, giving a possible combination of 256 selectable columns, which is just enough for the design of just 252 bitlines. There are 32 column-select lines available, each connected to a sense amplifier.

For row addressing, six address bits are used, while two additional ones are used for section addressing. The same input address lines are used for column and row addressing, therefore, the row addressing are latched into a register before decoding. The row decoding is also used to determine which of the three distinct waveforms is to be used in each of the reference and generate wordlines.

When valid data is available, a databus is selected from bits 1, 2 and 3 of the column-select address and bit 0 of the wordline select address. The following Fig-

ure 2.22 gives how the address bits are used for column and wordline addressing in ML3.

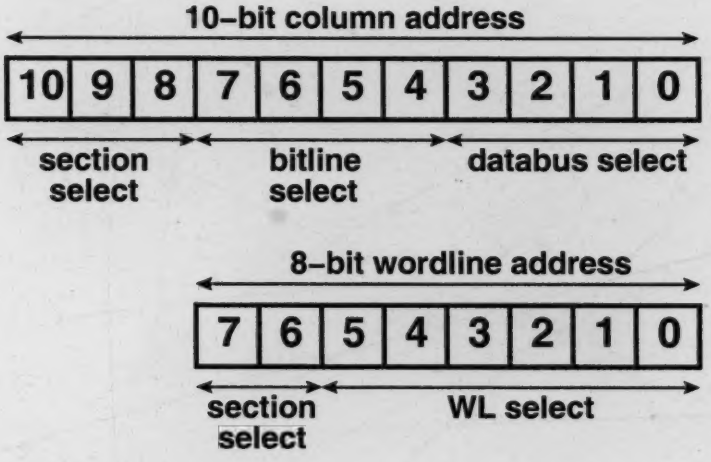


Figure 2.22: Row and column address bits in ML3

2.2.2.3 ML3 Chip Simulation and Verifications

ML3 was designed and simulated using Cadence design tools in 0.35-μm technology from TSMC.

An imperfection was found in the chip where the bitlines are supposed to be twisted close to the sense amplifiers in between the sections. Since the bitlines are not equally twisted there, as in the memory arrays, noise due to capacitive coupling on the bitlines cannot be avoided when the switch matrix or the sense amplifiers are activated [8].

Testing of one ML3 chip on the Agilent 81200 tester revealed a cell yield[9] of 3% for four-level operation. Nevertheless, much has been learned from this test chip. ML5 [42] (see Section 2.2.3, is a result of experience gained from the ML3 testing and the on-going research on MLDRAM at the University of Alberta.

[9]We define the cell yield to be the percentage of functional cells in the memory array of a particular chip over the total number of cells in the memory array.

2.2.3 Xiang's Variable-capacity MLDRAM

Chan's MLDRAM [8] is a direct realization of Birk's [5] four-level (two bits per cell) MLDRAM design in the form of a test chip. With a designed cell-capacity of only 2 bits per cell and the need for a switch matrix area, a sense amplifier for every sub-bitline and the extra decoding needed for multilevel sensing, a big disadvantage of Birk's MLDRAM is in the area overhead required. However, Birk's design shows promise in terms of noise immunity and speed. It is therefore worth investigating the expansion of Birk's MLDRAM scheme to more than four-levels. Increasing the cell capacity this way balances the economic disadvantage of the area overhead required and render it competitive in the DRAM market.

The expandable version of Birk's MLDRAM is ML5. Yunan Xiang has designed a variable-capacity MLDRAM based on Birk's design [42] in her thesis research work. The cell capacity is variable in that the required cell storage levels can be adjusted and the design is expandable to three, four and six levels. The five level mode in the test chip could not be tested due a mistake in the naming of the a signal going into a decoder.

Based on the idea that the reference voltage levels are evenly spaced to maximize the noise margin, an (N-1)-by-(N-1) array of sub-bitlines expanded from the original 3-by-3 array of Birk's design, can be used to extend the four-level operation, theoretically, to any number of operational levels. Recall that N is the number of analog data levels.

Xiang's MLDRAM is expandable to six levels, a 5-by-5 array should be used. Since it can be adjustable to two, three and four levels, the switch matrix voltages for charge-sharing for reference generation should be arranged in such a way that the 5-by-5 array for six-level operation can be resized and reused for other levels of operation. For four-level and three-level operation, the size of the reference generation switch matrix required is 3-by-3 and 2-by-2, respectively. For two-level DRAM mode operation, the switch matrix can be disabled and the bitline precharge value of $\frac{1}{2}V_{DD}$ be used as the reference value. Figure 2.23 shows the switch matrix voltage distribution configuration that can be used for the reference generation of

the three types of operational modes. Charge-sharing occurs along each section to create the required reference voltages.

VDC		Sections				
		A	B	C	D	E
Sub-bitline pairs	0	VSS	VSS	1/2 VDD	VSS	VDD
	1	VSS	VSS	VDD	VDD	VDD
	2	1/2 VDD	1/2 VDD	1/2 VDD	1/2 VDD	1/2 VDD
	3	VSS	VSS	VSS	VDD	VDD
	4	VSS	VDD	1/2 VDD	VDD	VDD

Figure 2.23: VDC reference generation voltage sources

ML5 [42] has similar parallel sensing scheme to ML3 [5, 8]. The difference in the architecture is that the switch matrix is bigger and more flexible to enable the expandable capability to six-level mode. The 2, 3, 4 and 6 levels of operation translates to 1, 1.5, 2 and 2.5 bits per cell capacity, respectively. In the switch matrix, the bitlines can be connected horizontally and vertically via transfer switches. There are six control signals pairs (for true and complement bitlines) from outside of the chip to form the 2-by-2, 3-by-3 and 5-by-5 arrays.

Figure 2.24 gives the test chip floorplan. As in ML3, the chip can be generally divided into the core and the periphery. The core consists of the memory array, sense amplifiers, databuses and wordline drivers. In the periphery, there are address decoders, databus decoders and IO buffers and latches. The RGX signal decoding and circuits is a legacy from ML3 and is used here in ML5. The RGX decoding circuits reside in the periphery of the test chip.

A simplified block diagram of the test chip is shown in Figure 2.25. In ML5, there are a total of five memory array sections. In each section, there are 16 wordlines, four of which are reference and generate wordlines. Therefore, only 12 are addressable wordlines connected to data storage cells. The number of bitlines built

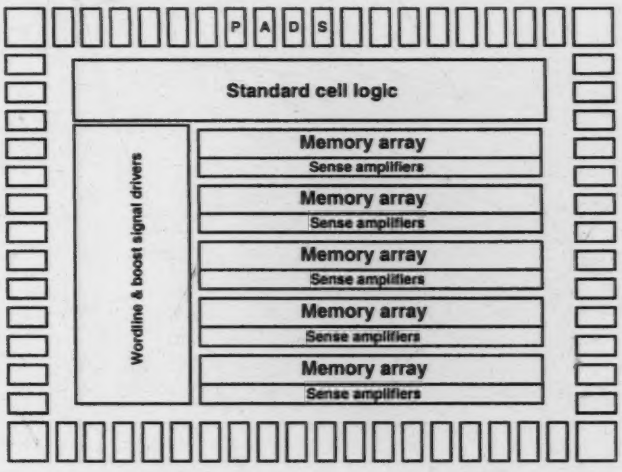


Figure 2.24: ML5 test chip floorplan

for the chip is 250. The bitlines are addressable via 32 databuses, so that every eight bitlines are allocated a databus line.

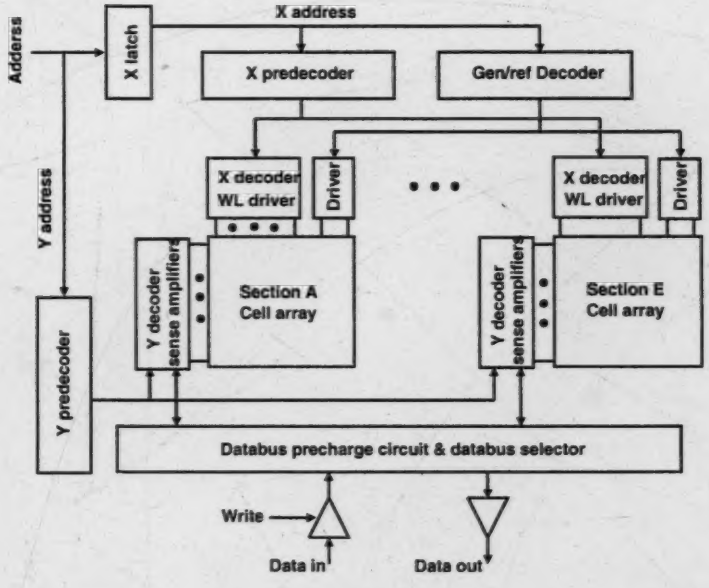


Figure 2.25: ML5 test chip simplified block diagram

2.2.3.1 ML5 Core

The ML5 core employs the folded bitline architecture without the twisting implemented in ML3. Friendly cells and reference and generate dummy cells are also used in the design. Since bitline twisting is not used, shielding is added in between bitline pairs to guard against bitline coupling.

The wordline drivers are boost circuits designed by ATMOS. The boost circuit is as shown in Figure 2.26. All wordlines are boosted along with the switches in the switch matrix.

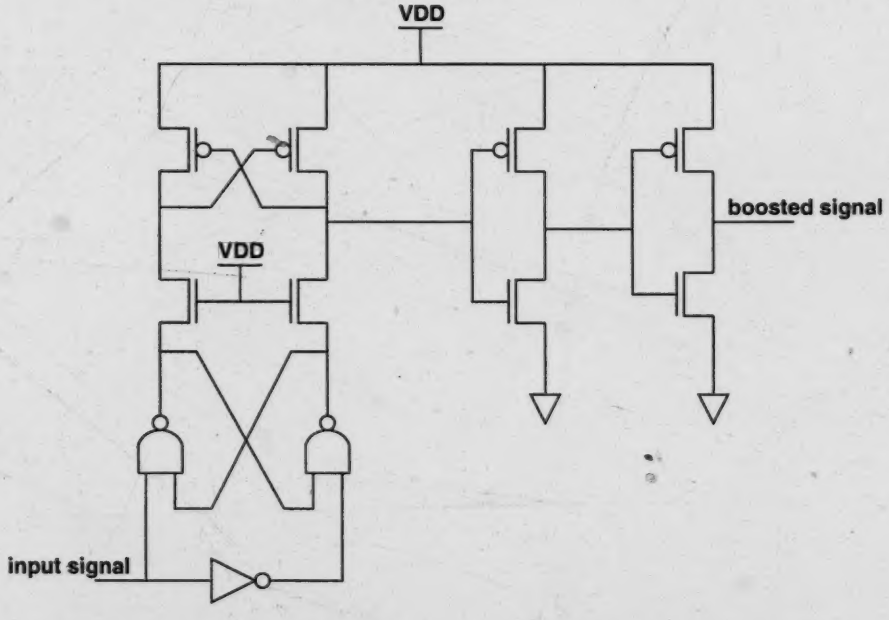


Figure 2.26: Wordline driver from ATMOS Corporation [42]

In terms of row and column addressing, ML5 has less complicated address scrambling since twisted bitlines were not used. However, it should be noted that the data read from odd wordlines are the opposite value of the data written.

2.2.3.2 ML5 Periphery

The address decoders, reference and generate wordline decoders and databus decoders all reside in the periphery area. The periphery is mostly made out of standard

cells from CMC's Cadence design tools library available in TSMC's 0.18-μm process technology. The reference and generate wordline (RGX signals) decoders are implemented as was done for ML3 [8] with some minor additions to accommodate reference and generate wordlines from the additional sections. There are three sections in ML3 while there are five in ML5 [42]. Figure 2.27 in the following shows how the address bits are used for address decoding. As in ML3 [8], the address bit for wordline decoding are latched into registers before they are pre-decoded, while the column addresses are not.

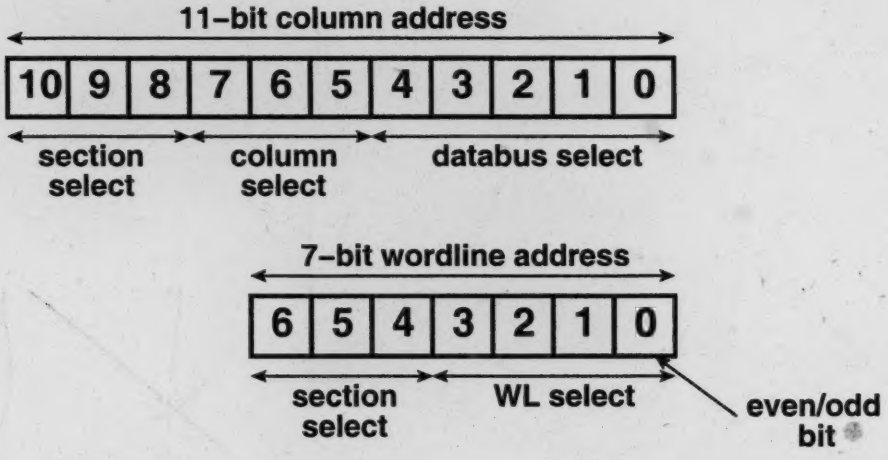


Figure 2.27: Row and column addressing bits in ML5

2.2.3.3 ML5 Chip Simulation and Verifications

The test chip was designed and laid out using Cadence design tools. ML5 chips were tested using the HP 81200 VLSI tester. The average yield for the ML5 chips was 78.5% for six-level operation. A significant improvement in the yield could be seen relative to the ML3 chips.

The next step in the research work would be to characterize the chip. ML5 was intended to be a characterization chip for Birk's MLDRAM design with the addition of variable capacity capability. Although the design has proven to be a significant improvement over the previous chip, ML5 lacks the visibility into the internal nodes of the chip, visibility that is so crucial for obtaining significant characterization

results. A major lesson learned while testing ML5 is that it is important to include circuitry that permits direct access to certain key internal signals. Measurements of leakage currents and internal voltage levels from the memory array and switch matrix will help answer many questions about the MLDRAM design.

The next chip in the MLDRAM series is ML6, which will be discussed in the chapters in the rest of the this thesis. ML6 is a more sophisticated characterization chip with built-in characterization circuits and internal voltage probes. It is also equipped with a temperature sensor to the core and periphery of the chip.

Chapter 3

The Design of ML6

The phenomenal growth in semiconductor memories over the past 30 years, particularly in DRAMs because of their function as main computer memories, has led to the investigation of new design methods that increase storage density and lower the cost per bit. The University of Alberta participated in this effort when the first multilevel DRAMs, ML2 and ML3, were developed by the VLSI research group. When ML3 was laid out in detail, it was found that Birk's MLDRAM design was at a great disadvantage when compared with conventional DRAM because of the additional area overhead. One of the ways to increase the storage density is to increase the number of levels per cell to more than four-levels per cell. ML5 was made to investigate this idea. The prototype of ML5 did work at all levels, although improvements were identified to make the design more robust and make the chip more suited for characterization work. The next step in the MLDRAM research is to implement a characterization chip with the benefit of the experience gained from designing and testing ML5 and ML3. ML6 is inspired by the need to directly monitor the internal operation of the MLDRAM circuitry. This characterization test chip is the focus of this thesis.

3.1 Test Chip Design Overview

As a characterization test chip, ML6 is equipped with built-in internal voltage probes designed to measure or observe signals on the internal nodes of the cir-

cuits. It also has special circuits to force binary signals onto bitlines and to overdrive databuses according to externally supplied analog voltages. Additionally, a temperature sensor[1] provides measurements from inside the chip as analog output voltages. To simplify and speed up address decoding during accesses[2] along a row of cells, the address decoder architecture is able to accommodate the automatic increment and decrement of the column and wordline addresses on-the-fly during testing, without having to reload the addresses from external inputs.

ML6 is a variable-capacity multilevel DRAM chip with essentially the same sensing and reference generation scheme used in the ML5 chip. It is ML5 made characterizable. The apparent design errors included in ML5 during the schematic and layout processes are removed from the design of ML6. One such fix is the ability to operate at five levels per cell. ML6 is able to function as a DRAM and is expandable to three-level, four-level, five-level and six-level modes. The complex address descrambling used in ML5 is avoided in ML6.

Some aspects of ML5 were modified and redesigned to increase noise immunity and robustness against charge-injection during switching. Transmission gate switches, instead of pass transistors, are used in the switch matrix and in the transfer switches in the bitlines in ML6. In the switch matrix, the addition of charge-injection cancellation transistors is unnecessary due to the use of transmission gates. The use of transmission gates on the transfer switches connected to the bitlines (e.g. the ISO switches connecting the sense amplifiers to the bitlines) avoids the need to boost these signals to a higher value than V_{DD}. There are no charge-injection cancellation transistors in ML5. Signal strength and fanout analysis was done to every input signal so that the correctly sized buffers are used. This step was not done properly for some of the signal drivers in ML5, which were strong enough to handle some of the larger signal fanouts present in the memory core (e.g the sense

[1] A temperature sensor is useful in an MLDRAM characterization chip in that the chip's operation with internal temperature changes can be observed. For an example, the variation of the memory cell data retention time with temperature can be observed with built-in temperature probes.

[2] Address bits are defined at external input pins to address the memory cells for data writing and retrieval. Accesses are occasions where the memory cells are addressed for writing of data and retrieval.

amplifier enable signals were never buffered appropriately in ML5).

The databus architecture is redesigned to include the differential read-write amplifier so that input data attenuation can be corrected before the data is driven onto the bitline and then onto the addressed cell. Having a differential amplifier on every databus pair (in addition to the bitline sense amplifiers) increases the robustness of the MLDRAM operations. The bitline and databus sense amplifiers can be made to drive the bitlines and the databus, respectively, at different times during sensing. Without the databus sense amplifiers, the bitline sense amplifiers would have to do all of the signal driving work from the bitline to the databus via the column select transistors. In ML5, a tri-state buffer is used to channel the data voltage from the bitline to the databus for reading and vice versa for writing.

The sense amplifier design was also modified to include a precharge circuit for the sense amplifier. This is in addition to the bitline precharge circuit that is already built in every bitline pair. In this way, the sense amplifiers and the bitlines can be isolated, precharged independently and used at different times. This extra degree of freedom is useful when we want to operate on the bitlines and the sense amplifiers differently. In ML5, there is only one precharge circuitry, which is shared by the sense amplifier and bitline.

The sense amplifier transistor sizes are also varied in ML6. This is so that it can be determined experimentally how differently sized sense amplifiers affect the drive power and offsets on the bitlines. There are two sets of sense amplifier sizes used in ML6 for this purpose. Since there are 320 bitlines in ML6, every set of forty bitlines are made to have one of the two sense amplifier sizes.

In addition to the different sense amplifier sizes, the cell sizes are also varied. Four cell sizes are used in the design. The pattern of varied cell sizes repeats for groups of eighty bitlines. The different cell sizes used in the ML6 chip allows information to be obtained on how different memory cell sizes affect the operation of the MLDRAM.

The power consumption of every circuit can be estimated through simulations using Cadence design tools. From the power estimations, the widths of the metals

line were calculated from their current density values taken from TSMC's 0.18-μm specifications, and used appropriately in the layout.

ML6 was designed using TSMC's CMOS 0.18-μm process technology. The core circuitry comprises mostly of analog circuits. They were designed by proceeding from schematic entry to physical layout and verified with analog simulation. Figure 3.1 shows the floorplan of the ML6. The core consists of custom layout components. The wordline boost and sense amplifier drivers are made as part of the memory core. The peripheral circuitry consists of the decoders, buffers and write drivers. The input pads, output pads and power pads are from TSMC's standard cell library tpz973g [37] while the other standard cells in the periphery are from the vst_n18_sc_tsm_c4 library [33, 34, 35, 36].

The CMOS 0.18-μm process technology that was used for ML6 is based on TSMC's proprietary 0.18-μm mixed-signal (digital and analog) dual-voltage process [36]. The libraries provided with the process contain standard cells which to the designer are black-box cells created based on the process design rules [34, 36] and verified using Simulation Program with Integrated Circuit Emphasis (SPICE) simulators and models [35]. The design rules for this process are available from TSMC's design rules and electrical parameters manual [34, 36]. The design rules are recommended for high yield and reliability logic and mixed-signal designs. The minimum width and length for a transistor in the process are 0.220 μm and 0.180 μm, respectively. The nominal power supply voltage for the 0.18-μm process is 1.8 V. However, the TSMC proprietary salicide process technology allows for power supply signals as high as 3.3 V.

Figure 3.2 shows the simplified block diagram of ML6. The core shows the five sections of the memory array, drivers and wordline boost drivers, sense amplifiers and switch matrix circuits for reference generation. The periphery shows the datapath from the address inputs to the decoders and to the memory core. From the memory core, the datapath continues to the databuses before ending at the output decoders as one binary data output signal, DATA_OUT.

There are in total 320 bitlines and five sections with 12 wordlines and 4 refer-

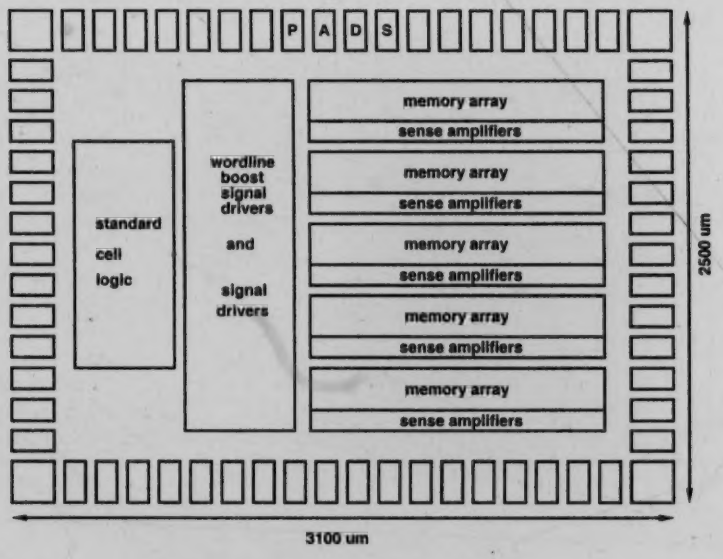


Figure 3.1: ML6 test chip floorplan

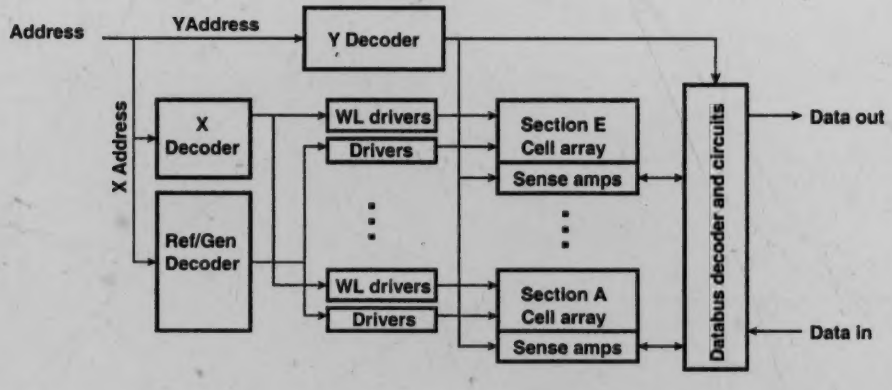


Figure 3.2: Simplified ML6 test chip block diagram

ence and generate wordlines per section. There are 40 databuses, with one databus for every 8 bitlines. Finally, there is one external pin each for data input and data output.

3.2 The Multilevel DRAM Array

The memory array contains the wordline drivers, memory cells, reference and generate circuitry, sense amplifiers, precharge circuitriy, and connection and isolation

switches. The connection and isolation switches are for connecting and isolating the sense amplifier with respect to the bitline and the bitline with respect to the databus. The wordline drivers and boost circuits fit into the wordline pitch. The memory cells, reference and generate cells, sense amplifiers, bitline switches such as the column-select and sense amplifier isolation switches, precharge circuits and switch matrix components (for reference voltage generation) fit into the bitline pitch.

These components in the memory array make up what is called the memory core. With 320 bitlines and 12 real wordlines connecting to data storage cells, the total number of addressable cells in each section is $320 \times 12 = 3840$ cells. Hence, the storage density of ML6 for all five sections is $5 \times 3840 = 19200$ cells. Figure 3.3 shows a simplified diagram of the memory array and its constituent components.

3.2.1 The Basic Memory Cell

The basic component of a DRAM array is the memory cell. The MLDRAM uses the 1T-1C memory cell as in a conventional two-level DRAM. The memory cell design for ML6 was adapted from the HDRAM cell from MOSAID Technologies Inc. that was used in test chips ML1 (an early MLDRAM attempt that functioned as a two-level DRAM), ML2 and ML3. In ML5, the storage node of the cell capacitor transistor is the connection between the source and the drain of the transistor as shown in Figure 3.4. The bulk voltage is -1 V. Since both the source and the drain voltages are higher than the bulk voltage most of the time, a back-gating (or body) effect occurs where the threshold voltage increases when the voltage between the source and body increases. Also, this method of storage increases leakage to the bulk from the cell since the storage node is at the channel of the cell capacitor transistor.

Figure 3.5 shows the basic cell for ML6. In this design, the cell plate and the substrate are connected together to provide back-biasing.[3] In ML6, the gate of the

[3]Back-biasing is to apply a negative voltage value to the subtrate so that an n-doped metal oxide semiconductor (NMOS) transistor can remain on even when the voltage applied to the gate is 0 V. In a memory cell, back-biasing the subtrate will ensure the storage capacitor is on all the time. This is need so that the data voltage of 0 V can be stored [20, 26].

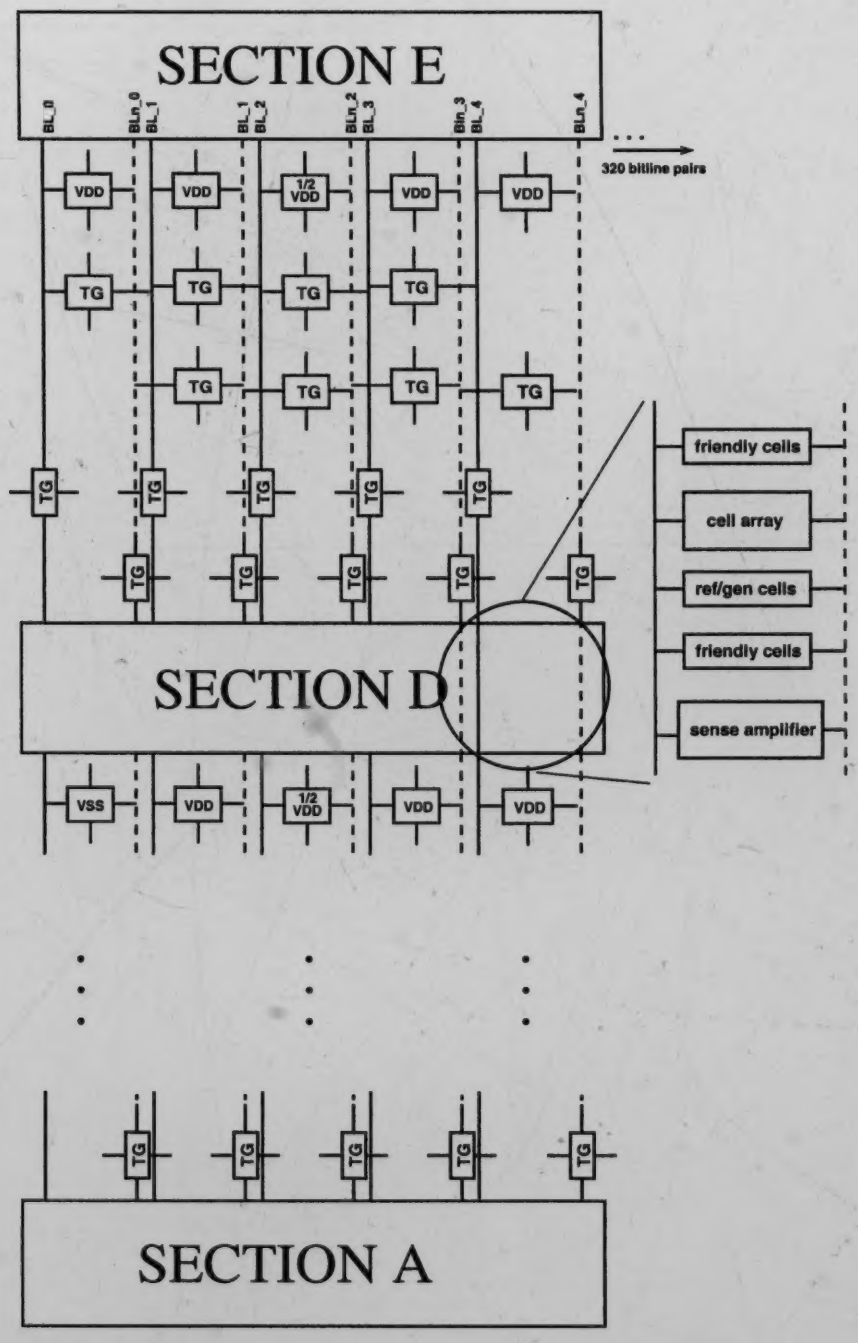


Figure 3.3: ML6 memory array showing the sections with five bitline pairs, switch matrix and columns

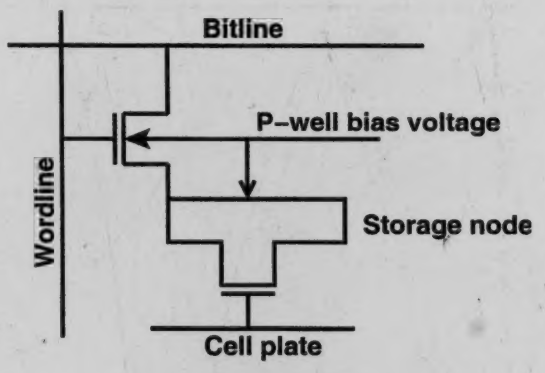


Figure 3.4: ML5 memory cell [43]

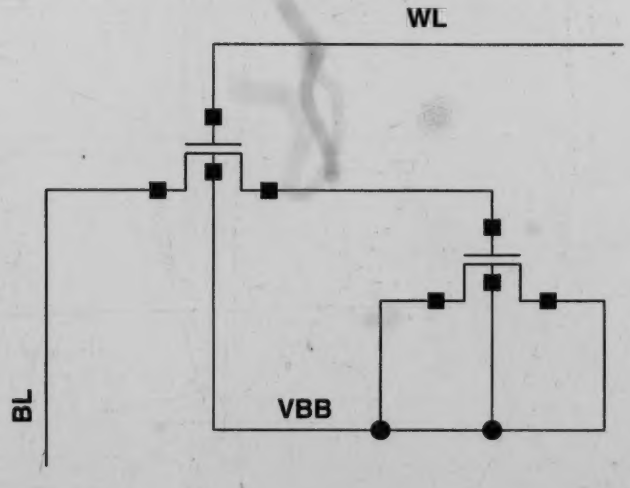


Figure 3.5: Schematic of the 1T-1C basic memory cell in ML6

storage transistor is used as the storage node instead of the transistor channel. This storage method avoids charge leakages from the channel — which is connected directly to the substrate — through the bulk connection [3, 26]. The selected back-bias voltage is $-1\,V$. The cell plate voltage is also set to $-1\,V$ to avoid back-gating (body-effect).[4]

The basic memory cells are arranged into sections in the manner shown in Figure 3.6. A memory array section consists of 320 bitline pairs, 12 addressable wordlines, 2 reference wordlines and 2 generate wordlines. The wordlines are connected

[4]The body-effect is when the source-bulk potential is higher than the substrate potential causing the threshold voltage of the transistor to be higher than V_{TH} [3, 25, 26].

alternately to real and complementary bitlines, implying that wordlines 0, 2, 4, 6, 8, 10 and 12 are connected to true bitlines, while wordlines 1, 3, 5, 7, 9, 11 are connected to complementary bitlines.

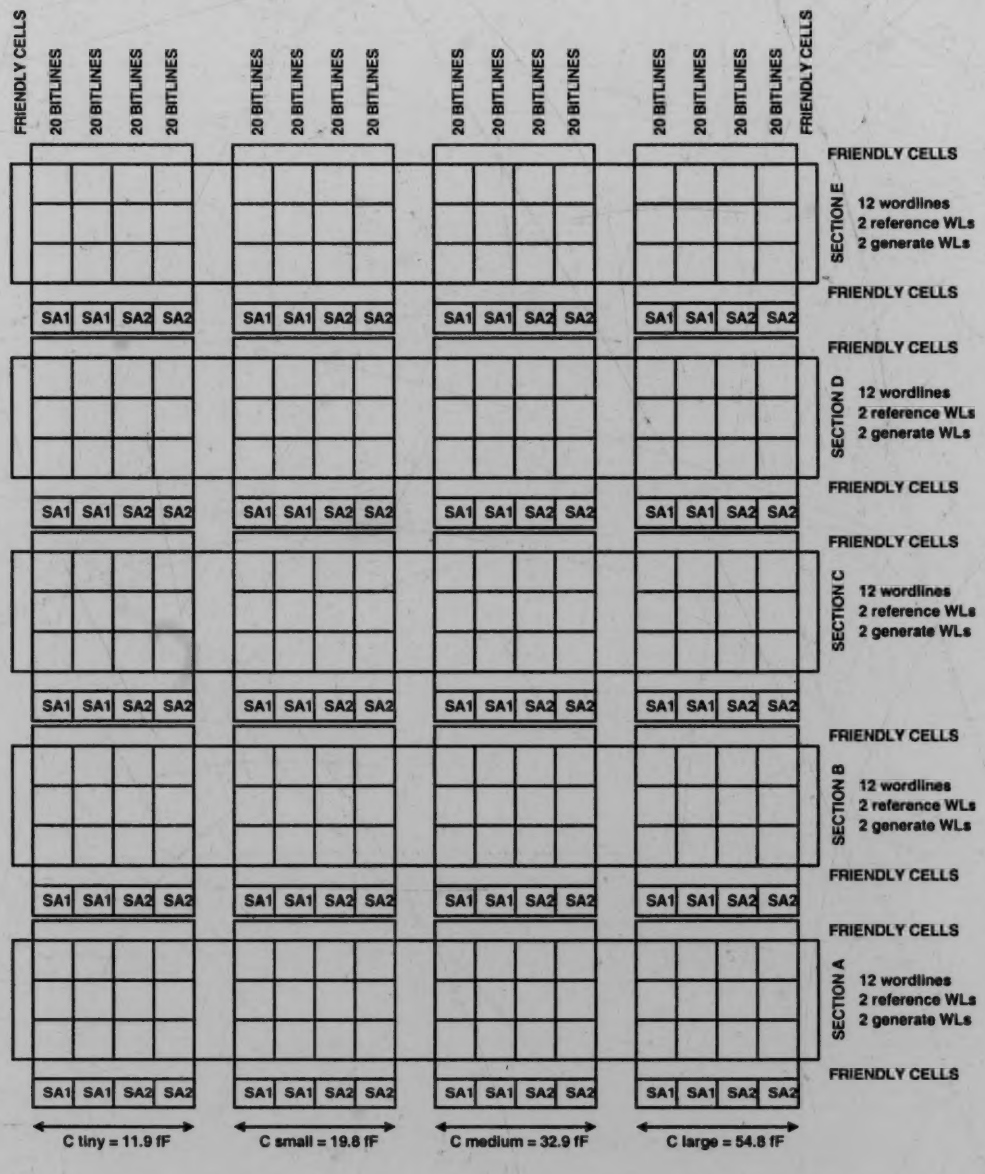
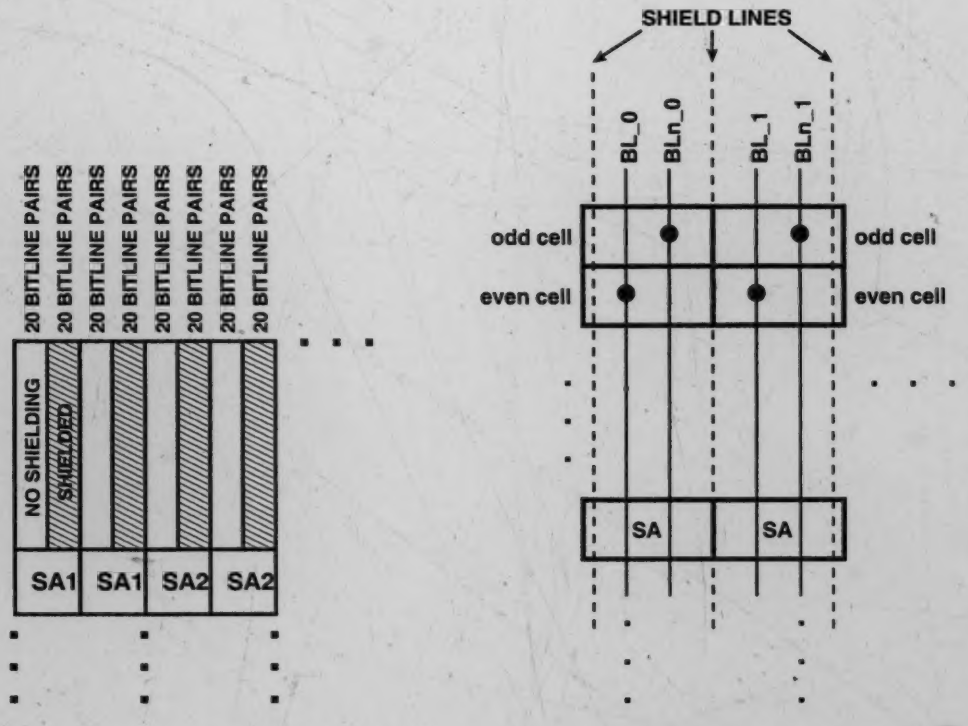


Figure 3.6: ML6 memory array showing cell arrangement and sizes

The reference and generate cells are exactly the same as the data storage memory cells. The reference cells are used to store the reference voltages for sensing,

while the generate cells are used to provide charge-balancing in the bitlines during reference generation and sensing operations. This method of charge-balancing was also used in ML3 and ML5. As in ML5, the bitlines are shielded from one another. Shield lines connected to the V_{SS} are laid out in between bitline pairs as an extra precaution for bitline coupling. The shielding lines run through the memory cells, reference-generate cells and sense amplifiers. Figure 3.7 shows how the shield lines are laid out alongside the bitlines. Only half of the bitline pairs with same sense amplifier sizing are shielded. This is so that we can observe the effectiveness of bitline shielding.



(a) Shielded-bitline distribution

(b) Shielding in between bitline pairs

Figure 3.7: Shielding in between sub-bitlines in ML6

The bitlines are not long enough to benefit from the twisted architecture since there are only 16 wordlines in each section. The cell sizes in the array range from tiny (11.9 fF), to small (19.8 fF), medium (32.9 fF) and large (54.8 fF). Starting

from the zeroth bitline, every eightieth bitline sees a cell size increase.

The memory array is surrounded by one layer of unused friendly cells so that the electrical environment seen by all the data-storing memory cells is the same. Figure 3.6 shows the layout of the friendly cells in the array. The access transistors to the friendly cells are connected permanently OFF. Hence, the gates of the friendly cells are either tied to V_{DD} or V_{SS}. The friendly cells are not used to store data.

The reference and generate dummy cells are similar to the data-storing memory cells except that they have the gates of their cell access transistors omitted. The dummy cells can be accessed to balance out charge injection on the bitlines when data-carrying cells are accessed. The dummy cells on the reference wordlines are used to store reference voltages while the cells on the generate wordlines are used for charge cancellation and capacitance balancing. During sensing, the addressed wordline is asserted for cell access. The bitline segment with the addressed cell would experience capacitance from the bitline plus one memory cell. To achieve capacitive balance in the sensing array for multilevel sensing, selected reference and generate cells are asserted and the bitlines in the switch matrix are appropriately connected. This is done so that other bitline segments in the sensing array will experience the same capacitance as the bitline segment with the addressed cell.

3.2.2 The Reference and Generate Cells

The reference and generate cells are placed with the memory cells as shown in Figure 3.8. This design was also used in ML3 and ML5. There are two types of sub-bitlines: the reference sub-bitlines and the generate sub-bitlines.

As shown in Figure 3.8 the reference (R) cells are organized among the generate (G) cells in the following order in the memory array: G G R G G.

The operations of the reference and generate wordlines are explained in Chapter 4. Figure 3.9 shows the ML6 reference generation scheme for 2 to 6-level operation.

The reference voltages are generated by charge sharing voltages on the bitlines. Six-level operation uses a 5-by-5 array of sub-bitlines, while 5-level operation uses a 4-by-4 array, and 3-level operation uses a 2-by-2 array. For two-level DRAM op-

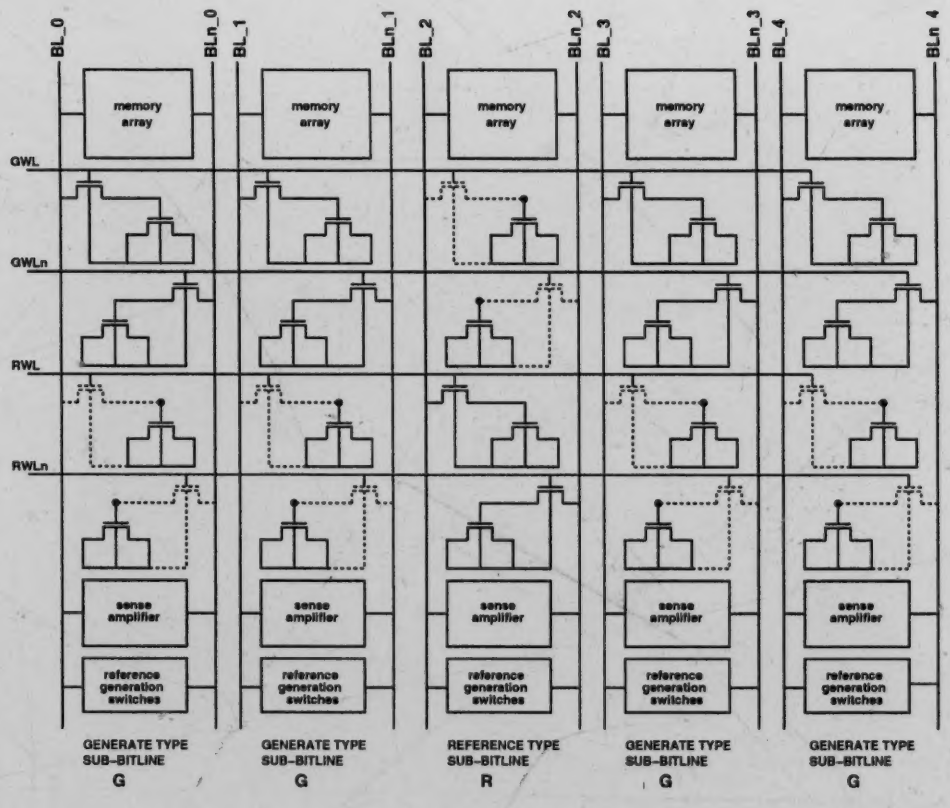


Figure 3.8: ML6 sub-bitline types and arrangement

eration, the reference generation technique is not needed since the precharge voltage is already set at $\frac{1}{2}V_{DD}$. The precharge voltage can be used in this last case as the reference voltage.

The connections and reference generation actions are controlled by a switch matrix, as shown in Figure 3.3. The switch matrix bitlines are shielded to avoid capacitively coupled noise around the time of sensing.

The switch matrix is not used in reference generation for two-level operation because the bitline precharge voltage of $\frac{1}{2}V_{DD}$ is already the reference voltage.

From the 2-by-2 array shown in Figure 3.9 of the switch matrix formed by sections B and C, and bitlines 1 and 2, the following calculations can be made to obtain the reference voltages for three-level operation. During reference generation, the bitlines are precharged to the indicated values before being connected together

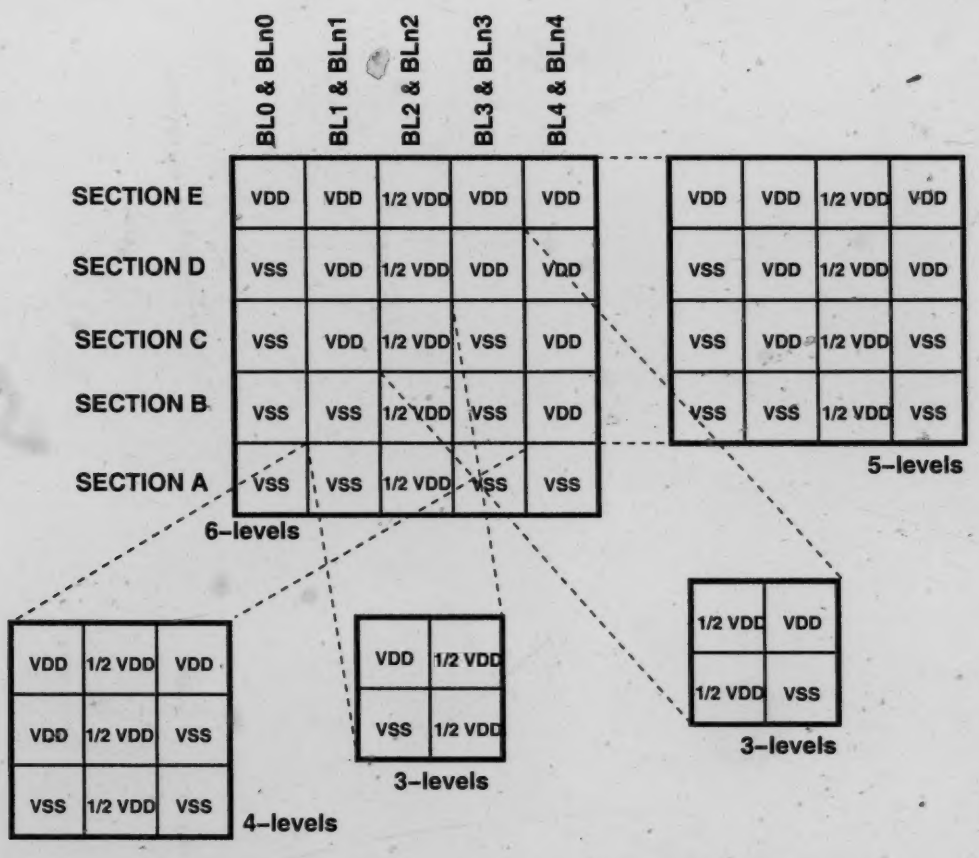


Figure 3.9: ML6 switch matrix analog voltages for reference generation

and charge shared to create the correct reference values. Equations 3.3 and 3.6 show how the two voltages (V_{SS} and $\frac{1}{2}V_{DD}$) are shared to generate the appropriate reference voltages. The calculations shown are for sections B and C, and across bitlines 1 and 2 in the switch matrix. Sections C and D, across bitlines 2 and 3 can also be used to generate the reference voltages.

In Section B, assuming three-level operation, we have across bitlines 1 and 2:

$$V_{REF1} = \frac{V_{SS} + \frac{1}{2}V_{DD}}{2} \tag{3.1}$$

$$= \frac{1}{4}V_{DD} \tag{3.2}$$

$$= 0.45\,V \quad \text{if } \cdot V_{DD} = 1.8\,V \tag{3.3}$$

In Section C across bitlines 1 and 2 we have:

$$V_{REF2} = \frac{V_{DD} + \frac{1}{2}V_{DD}}{2} \tag{3.4}$$

$$= \frac{3}{4}V_{DD} \tag{3.5}$$

$$= 1.35\,V \quad \text{if} \quad V_{DD} = 1.8\,V \tag{3.6}$$

For four-level operation, only one set of switch matrix configurations can be used for reference generation. From Figure 3.9, the combination of voltages from sections B, C and D, across from bitlines 1, 2 and 3, can be computed to form the three required reference voltages.

In Section B across bitlines 1, 2 and 3:

$$V_{REF1} = \frac{0 + \frac{1}{2}V_{DD} + 0}{3} \tag{3.7}$$

$$= \frac{1}{6}V_{DD} \tag{3.8}$$

$$= 0.30\,V \quad \text{if} \quad V_{DD} = 1.8\,V \tag{3.9}$$

In Section C across bitlines 1, 2 and 3:

$$V_{REF2} = \frac{V_{DD} + \frac{1}{2}V_{DD} + 0}{3} \tag{3.10}$$

$$= \frac{3}{6}V_{DD} \tag{3.11}$$

$$= 0.90\,V \quad \text{if} \quad V_{DD} = 1.8\,V \tag{3.12}$$

In Section D across bitlines 1, 2 and 3:

$$V_{REF3} = \frac{V_{DD} + \frac{1}{2}V_{DD} + V_{DD}}{3} \tag{3.13}$$

$$= \frac{5}{6}V_{DD} \tag{3.14}$$

$$= 1.50\,V \quad \text{if} \quad V_{DD} = 1.8\,V \tag{3.15}$$

For five-level operation, there are two sets of voltage combinations that can be used for reference voltage generation. The two corresponding 4-by-4 arrays consist of sections B, C, D and E, across bitlines 0, 1, 2 and 3, and sections A, B, C and D, across bitlines 1, 2, 3 and 4. Any one of the 4-by-4 arrays will produce the correct

reference voltages for five-level operation. The following equations show how the reference voltages can be obtained from charge-sharing in the switch matrix using sections B, C, D and E, across bitlines 0, 1, 2 and 3.

In Section B across bitlines 0, 1, 2 and 3:

$$V_{REF1} = \frac{0+0+\frac{1}{2}V_{DD}+0}{4} \quad (3.16)$$

$$= \frac{1}{8}V_{DD} \quad (3.17)$$

$$= 0.225\,V \quad \text{if} \quad V_{DD} = 1.8\,V \quad (3.18)$$

In Section C across bitlines 0, 1, 2 and 3:

$$V_{REF2} = \frac{0+V_{DD}+\frac{1}{2}V_{DD}+0}{4} \quad (3.19)$$

$$= \frac{3}{8}V_{DD} \quad (3.20)$$

$$= 0.675\,V \quad \text{if} \quad V_{DD} = 1.8\,V \quad (3.21)$$

In Section D across bitlines 0, 1, 2 and 3:

$$V_{REF3} = \frac{0+V_{DD}+\frac{1}{2}V_{DD}+V_{DD}}{2} \quad (3.22)$$

$$= \frac{5}{8}V_{DD} \quad (3.23)$$

$$= 1.125\,V \quad \text{if} \quad V_{DD} = 1.8\,V \quad (3.24)$$

In Section E across bitlines 0, 1, 2 and 3:

$$V_{REF4} = \frac{V_{DD}+V_{DD}+\frac{1}{2}V_{DD}+V_{DD}}{4} \quad (3.25)$$

$$= \frac{7}{8}V_{DD} \quad (3.26)$$

$$= 1.575\,V \quad \text{if} \quad V_{DD} = 1.8\,V \quad (3.27)$$

For six-level operation, the full 5-by-5 switch matrix is used for reference generation. The reference voltages can be generated from the full 5-by-5 switch matrix as shown in the following equations. All the sections, A, B, C, D and E, across bitlines 0, 1, 2, 3 and 4, make up the 5-by-5 switch matrix for six-level operation.

In Section A across bitlines 0, 1, 2, 3 and 4:

$$V_{REF1} = \frac{0+0+\frac{1}{2}V_{DD}+0+0}{5} \quad (3.28)$$

$$= \frac{1}{10}V_{DD} \quad (3.29)$$

$$= 0.18\,V \quad \text{if} \quad V_{DD} = 1.8\,V \quad (3.30)$$

In Section B across bitlines 0, 1, 2, 3 and 4:

$$V_{REF2} = \frac{0+0+\frac{1}{2}V_{DD}+0+V_{DD}}{5} \quad (3.31)$$

$$= \frac{3}{10}V_{DD} \quad (3.32)$$

$$= 0.54\,V \quad \text{if} \quad V_{DD} = 1.8\,V \quad (3.33)$$

In Section C across bitlines 0, 1, 2, 3 and 4:

$$V_{REF3} = \frac{0+V_{DD}+\frac{1}{2}V_{DD}+0+V_{DD}}{5} \quad (3.34)$$

$$= \frac{5}{10}V_{DD} \quad (3.35)$$

$$= 0.9\,V \quad \text{if} \quad V_{DD} = 1.8\,V \quad (3.36)$$

In Section D across bitlines 0, 1, 2, 3 and 4:

$$V_{REF4} = \frac{0+V_{DD}+\frac{1}{2}V_{DD}+V_{DD}+V_{DD}}{5} \quad (3.37)$$

$$= \frac{7}{10}V_{DD} \quad (3.38)$$

$$= 1.26\,V \quad \text{if} \quad V_{DD} = 1.8\,V \quad (3.39)$$

In Section E across bitlines 0, 1, 2, 3 and 4:

$$V_{REF5} = \frac{V_{DD}+V_{DD}+\frac{1}{2}V_{DD}+V_{DD}+V_{DD}}{5} \quad (3.40)$$

$$= \frac{9}{10}V_{DD} \quad (3.41)$$

$$= 1.62\,V \quad \text{if} \quad V_{DD} = 1.8\,V \quad (3.42)$$

3.2.3 The Bitline Sense Amplifier

The sense amplifier design in ML6 fits within the bitline pitch (which is 4.2 μm) and is shielded from the bitlines to reduce switching noise. The bitline pitch was itself

largely determined by the size of the largest cell capacitors, which were 4 μm by 32 μm. Figure 3.7 shows how the sense amplifers are shielded in the layout. In ML6, the sense amplifier circuitry contains the following components: sense amplifier precharge circuitry, sense amplifier, column select transistors, isolation transistors and bitline precharge circuitry. There is one sense amplifier circuit for every bitline pair. In addition, one databus line is allocated to every 8 sense amplifiers. The bitline segment of the addressed databus connects to the appropriate databus via column-select transistors. Altogether, 40 databus lines fan out to 320 bitlines in the array.

The sense amplifier precharge and bitline precharge circuits precharge the sense amplifier and bitline pair to $\frac{1}{2}V_{DD}$. The sense amplifier precharge transistors are minimum size transistors. They can be minimum-size since the sense amplifier precharge circuit only needs to precharge the small sensing node capacitance of the sense amplifiers. Due to the addition of isolation transistors to the bitlines, the bitlines and sense amplifiers can be precharged independently. The drive that is needed from the sense amplifier precharge circuit is minimal. The bitline precharge transistors are made bigger since the bitlines are generally longer compared to the bitline sections connected to the sense amplifier. Figure 3.10 illustrates the components of the sense amplifier circuit used in ML6. The sense amplifier-bitline isolation transistors, however, are transmission gates so that full rail-to-rail voltages can be transfered from the bitlines to the sense amplifiers. The transmission gate isolation transistors were made minimum size.

The sense amplifier block consists of two back-to-back inverters connected as shown in Figure 1.2. The figure is repeated in Figure 3.11 for convenience. The PMOS and NMOS transistors in the sense amplifier block are made the same minimum size. Two different sense amplifier sizes are included in the ML6 design. Type 1 has a channel length of $L = 360nm$ and a channel width of $W = 1.0\mu m$ while type 2 has a channel length of $L = 480nm$ and a channel width of $W = 1.0\mu m$. The widths are kept the same while the lengths are changed. Two sense amplifier sizes were included to permit experiments that explore how different sizes of sense

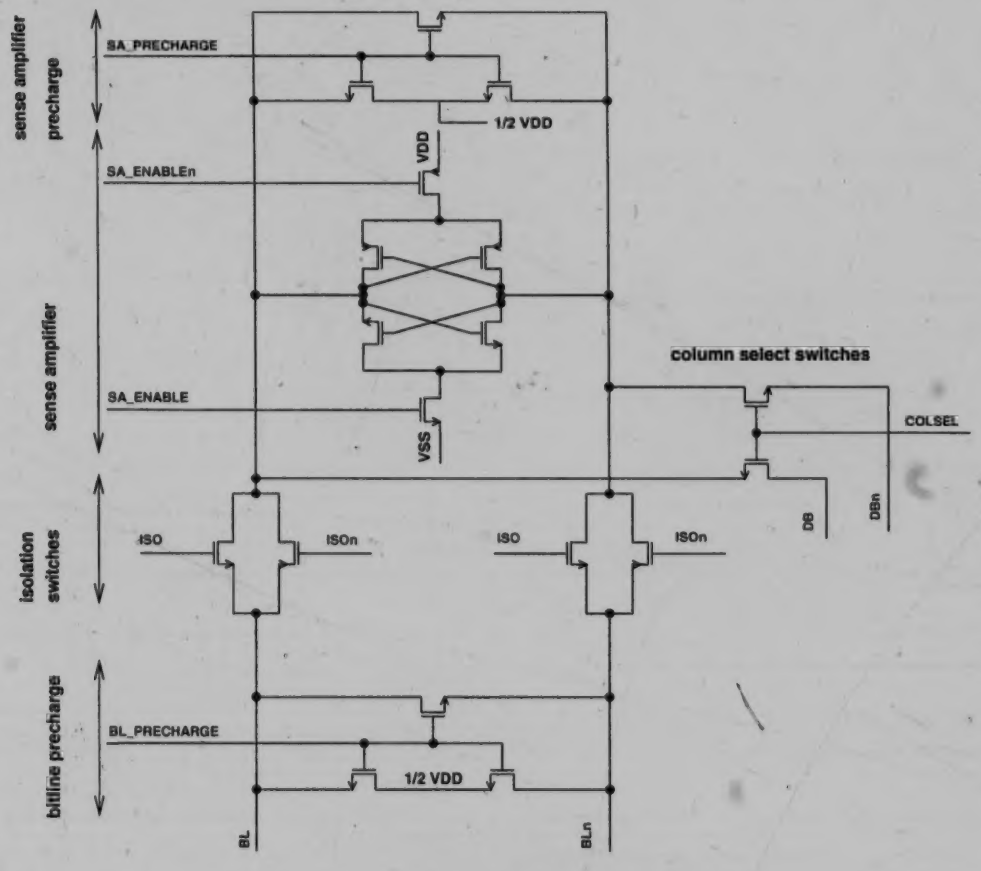


Figure 3.10: ML6 sense amplifier circuit

amplifier operate with different memory cell sizes. Every forty bitine pairs is alternated with one of the sense amplifier sizes. This arrangement is shown in Figure 3.6.

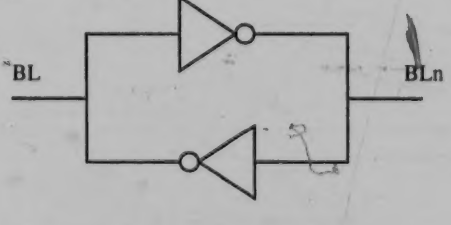


Figure 3.11: Back-to-back inverters

3.2.4 The Databus and Read-Write Sense Amplifier

The same sense amplifier design is also used to interconnect databus segments which provide a signal path from the bitlines to the write driver and pads. Figure 3.12 shows the databus sense amplifier for reading and writing the databus. The databus read/write sense amplifier circuitry consist of the precharge circuit, pull-up and pull-down circuitry and output buffer for the DATA_OUT signal to the pads. Included in the databus circuitry is also the write driver which has more drive than an ordinary sense amplifier.

The databus is precharged to V_{DD} . When DATA_IN is 0 and DB_WRITE_EN is asserted, the complement databus will stay at the precharge voltage of V_{DD} while the true databus is discharged to $0V$. When DATA_IN is 1, the opposite happens. That is, the complement databus will be discharged to $0V$ while the true databus stays at the precharge voltage V_{DD}. Meanwhile, the true databus experiences a voltage drop which will be pulled down to $0V$ by the sense amplifier. The opposite happens when the DATA_IN is 1. The architecture of the databus sense amplifier requires that DB_SA_EN be switched on at the correct time for a successful read operation. If the DB_SA_EN signal is asserted before the signal on the bitlines is transfered onto the databus by the column select signal, incorrect data could be read. One databus line is dedicated to every 8 sense amplifiers. Altogether, 40 databus lines fan out to 320 bitlines in the array.

3.2.5 The Signal and Wordline Boost Drivers

The wordlines have to be boosted to above V_{DD} since these signals control n-type cell access transistors which need to pass both V_{DD} and V_{SS} signals without attenuation. The isolation transistors have to be large enough so that the voltage drop from the bitline to the memory cells or vice versa, is minimum. Figure 3.13 shows a boost circuit design provided to us by ATMOS. It is used as a wordline boost driver circuit. The circuit was used in ML5 (also shown in Figure 2.26 of Chapter 2 Section 2.2.3.1) and then modified and reused, with minor changes in ML6. An inverter is added to create the INPUTn signal. Two additional inverters are also

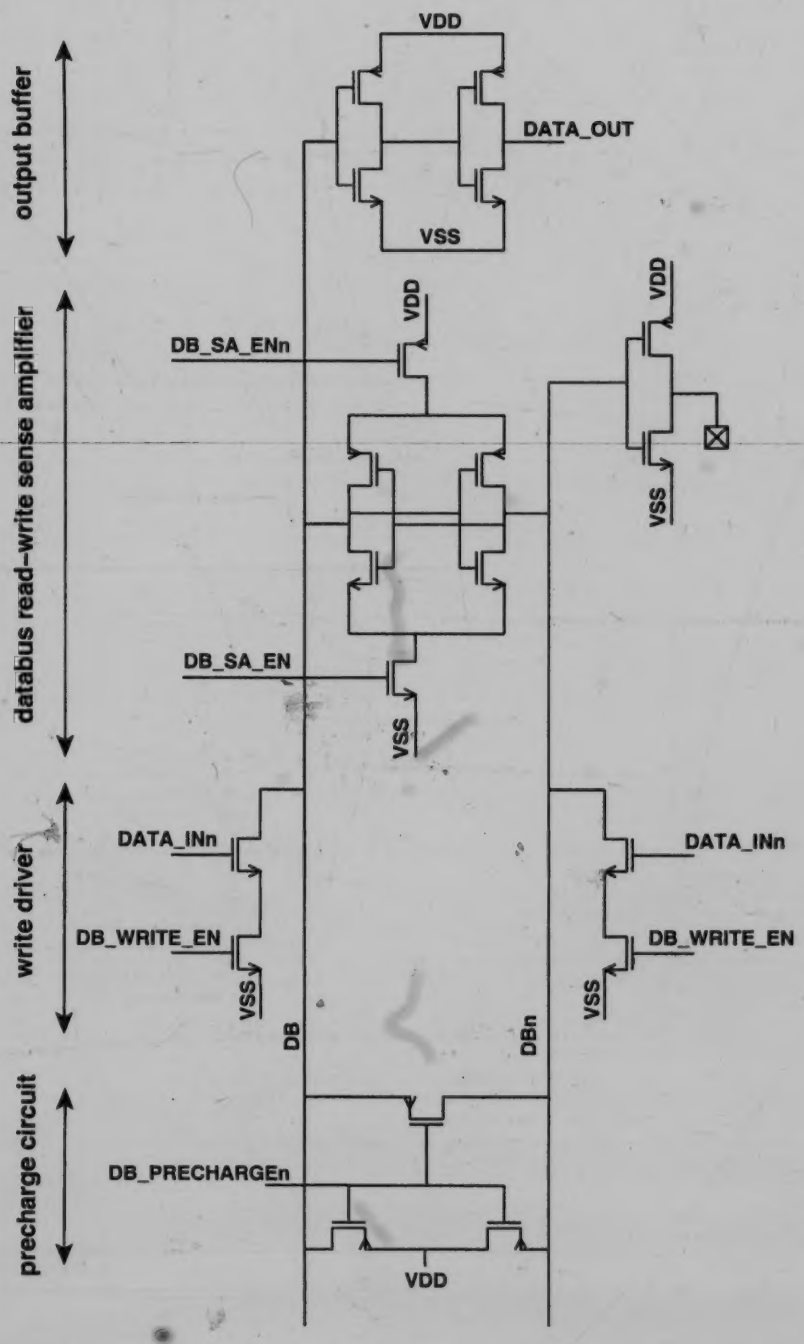


Figure 3.12: ML6 databus read-write sense amplifier circuitry

added before the output signal to amplify output current drive with minimal delay. The modified wordline boost circuit used in ML6 is shown in Figure 3.14.

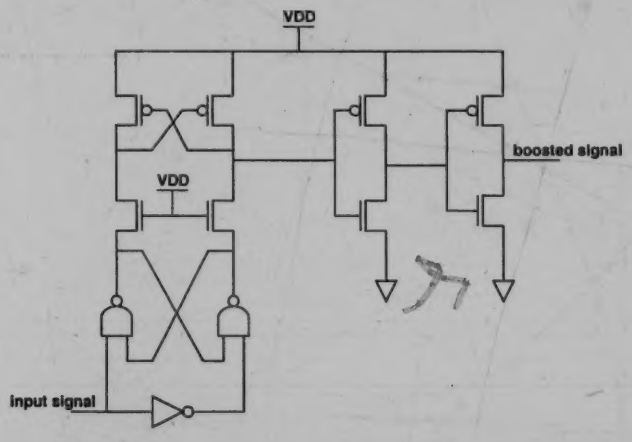


Figure 3.13: ML5 wordline driver from ATMOS Corporation [42]

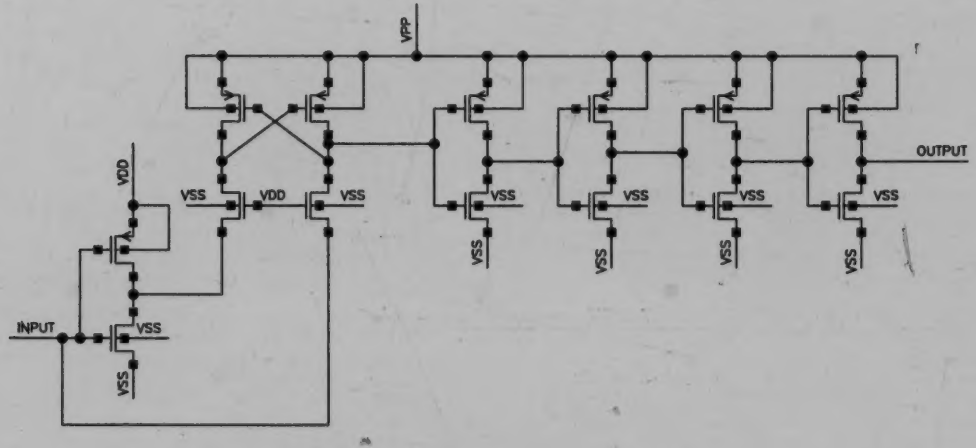


Figure 3.14: ML6 wordline boost driver

The signals going into the sense amplifier and switch matrix for reference generation are also buffered to provide sufficient drive current.

3.3 The Peripheral Circuitry

The peripheral circuitry consists mostly of decoders, buffers, descrambling and scrambling circuits. In ML6, only about 10% of the layout area is occupied by

Table 3.1: ML6 Multiplexed Address Control Opcodes

Opcode Name	Address Decode Select	Description
NOP	0000	hold everything
INC_X	0001	increment the X address
LOAD_X0	0010	load lower part of the X address (X_ADDR[3:0])
LOAD_X1	0011	load upper part of the X address (X_ADDR[6:4])
DEC_X	0100	decrement the X address
RESET_X	0101	resets only the X address
INC_Y	0111	increment the Y address
LOAD_Y0	1000	load lower part of the Y address (Y_ADDR[3:0])
LOAD_Y1	1001	load middle part of the Y address (Y_ADDR[7:4])
LOAD_Y2	1010	load upper part of the Y address (Y_ADDR[11:8])
DEC_Y	1011	decrement the Y address
RESET_Y	1100	reset only the Y address
RESET	1111	reset both X address and Y address

the peripheral circuitry.

3.3.1 Address and Reference-Generate Signal Decoders

The address decoders are divided into two main categories: the X decoders are row decoders while the Y decoders are column decoders. The X and Y decoders are both created by synthesizing Verilog code.

There are four address inputs and four address-decoder control inputs. The four address-decoder control inputs are used to control and latch the four address inputs for appropriate operations on the input address bits. Table 3.1 shows the opcode table for the multiplexed control bits.

There are 7 X-decoder register bits to accommodate the addressing of the 5 sections and 12 wordlines. Register bit XADDR[0] determines whether an even or odd wordline is selected. Register bits XADDR[0:3] are used for wordline decoding while bits XADDR[4:6] are used for section decoding. Since there are only four external address inputs, the addresses must be multiplexed.

For Y or bitline decoding, 12 register bits are used. Bits YADDR[0:2] are used for section decode. Bit YADDR[3-8] are used for databus selection while YADDR[9:11] are used for column or bitline selection. Similar to the X address decoding, the Y address inputs have to be multiplexed to be loaded (in three clocked steps) into the 12 register bits.

The addresses are latched into the respective registers by a clock pulse. Other operations that can be performed by the X and Y decoders are the increment and decrement of the addresses for fast page mode operations along the same accessed row of cells. When operations on the X and Y addresses are active, the other addresses are held unchanged until the next opcode for a new operation is received. The Verilog code is also written in such a way that NOPs, RESETs and invalid addresses set the active address to '00..00' while the contents of the other address register remain unchanged.

Figure 3.15 gives the row and column address mapping in ML6. The row and column addresses map directly to the physical positions of the cells without scrambling since there is no bitline twisting.

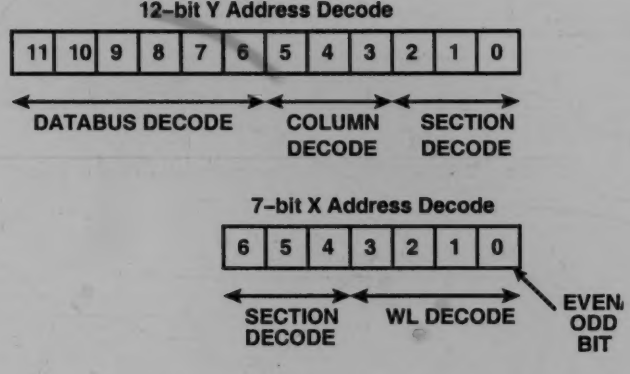


Figure 3.15: ML6 X and Y address fields bits

The reference and generate wordline decoders are also set in the periphery. The reference and generate wordlines are switched on and off according to specific waveforms, which depend on the wordline and bitline addressed. The addresses determine the waveforms on the reference and generate wordlines during each operation. The reference-generate signal decoder uses addressing information to gen-

erate the appropriate reference and generate signals sent to the X decoder. Then, the X decoder uses the decoded reference and generate signals (RGX1, RGX2, RGX3), to generate the final set of reference and generate signals for each section.

In addition to the Verilog-coded address decoders, there is also built-in address descrambling circuitry. The descrambling XOR gates were not included in ML5 and this caused extra work to descramble the test vector data during testing. In ML6, bit XADDR[0] determines whether the accessed wordline is even or odd. Both the input data and output data are put through XOR gates with bit XADDR[0] so that the descrambling can be done for both data input and output. When bit XADDR[0] is a 0, an even wordline is accessed, so the data remains the same for the input and output. But when bit XADDR[0] is a 1, the data is flipped for both the input and output signal paths. The descrambling circuit ensures that the output data is not flipped when odd memory cells are accessed.

3.4 Built-in Designed-for-Characterization Circuits

In addition to including variable cell and sense amplifier sizes, some circuits are built into ML6 to aid in the characterization of the MLDRAM. These circuits are probing circuits that allow the voltages on internal nodes to be brought out of the chip for external measurements.

3.4.1 Databus Analog-Voltage Set Circuit

This circuit is able to accept an external analog voltage and then drive it onto all of the databus lines. Figure 3.16 shows the schematic of the analog-voltage-set circuitry.

The purpose of this circuit is so that one can force voltages onto the databus to check for defects in the databus as well as the databus sense amplifier input-output circuits.

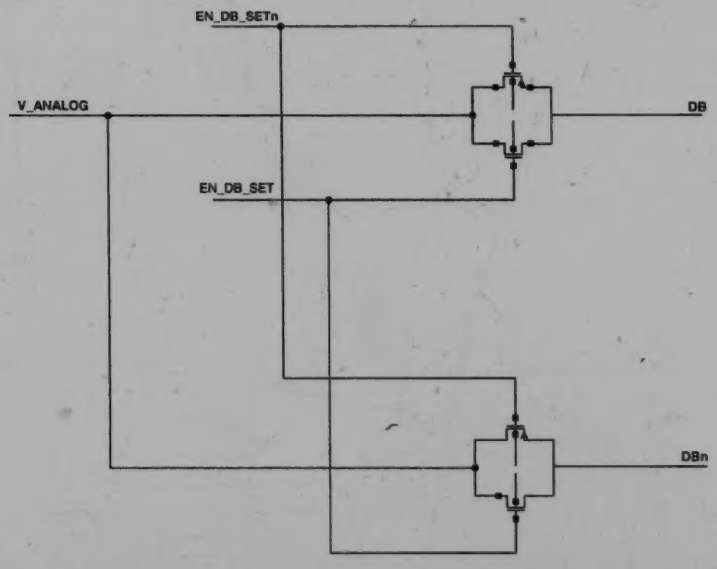


Figure 3.16: ML6 databus analog-voltage set circuit

3.4.2 Analog-to-Digital Probe Circuit

The analog-to-digital probe circuitry allows internal analog voltage measurements to be accessible as digital outputs. To do this, there is a differential amplifier that compares an external analog reference voltage to the internal analog voltage. So, the digital output can be switched by varying the reference voltage. If the digital output is high, it can be concluded that the internal analog probe voltage is higher than the reference voltage. If the digital output is low, the internal analog voltage probe lower than the reference voltage. Figure 3.17 shows the schematic of one of the probe points.

There are 62 probe points in the memory core. A 63rd probe point is an external input so that we can compare the external and reference voltages to verify the functionality of the probing circuit. The 63 probe points are selected using a set of external pins. The probe points are selected by sending the appropriate probe number to the memory core. The appropriate number is generated using a clock signal that pulses the correct number of times through a shift register. The probe connections in the memory core are listed in a table in Appendix E.

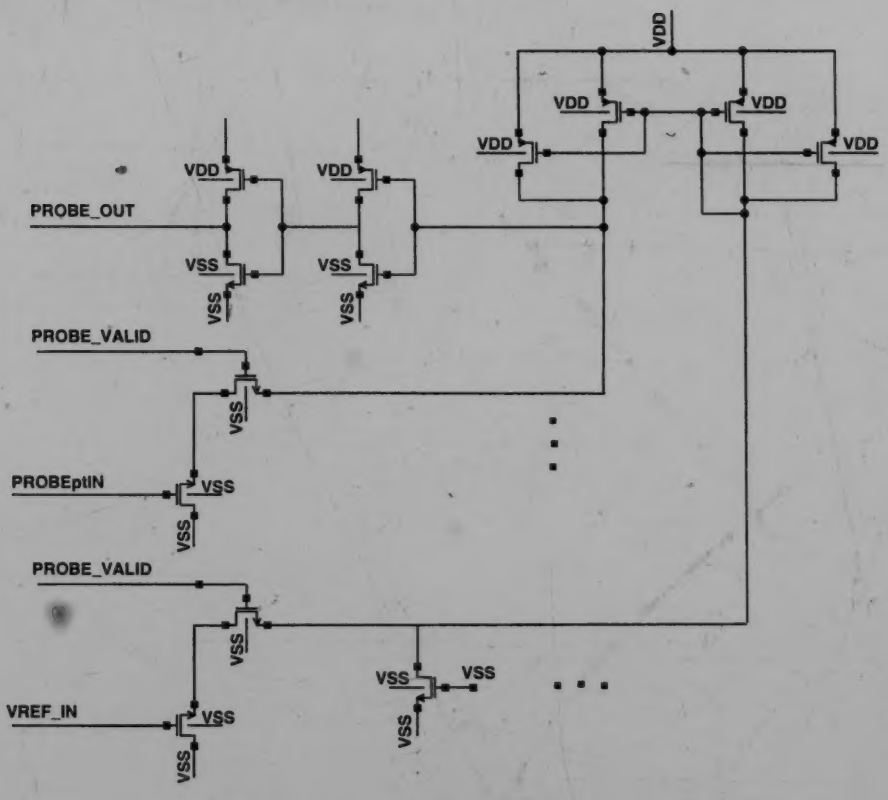


Figure 3.17: ML6 analog-to-digital probe

3.4.3 Temperature Probe Circuit

Two temperature probes were included in the memory core and periphery so that the operating temperature could be measured. The output of both probes is analog and there is no external input to the chip for the operation of the temperature probes. The temperature probe circuitry is shown in Figure 3.18 [31].

The temperature probe is essentially a current differential amplifier that reacts to temperature changes and sends a voltage change to the output [4, 28, 29, 38]. Since the outputs to the temperature probes are analog, they have to be connected directly to an oscilloscope (or a voltmeter) for observation.

This temperature probe design is chosen because it is small (nine transistors) and it does not use bipolar junction transistors (BJTs) as most analog temperature probes do. A parametric sweep of the sensing temperature range is run from 0 to

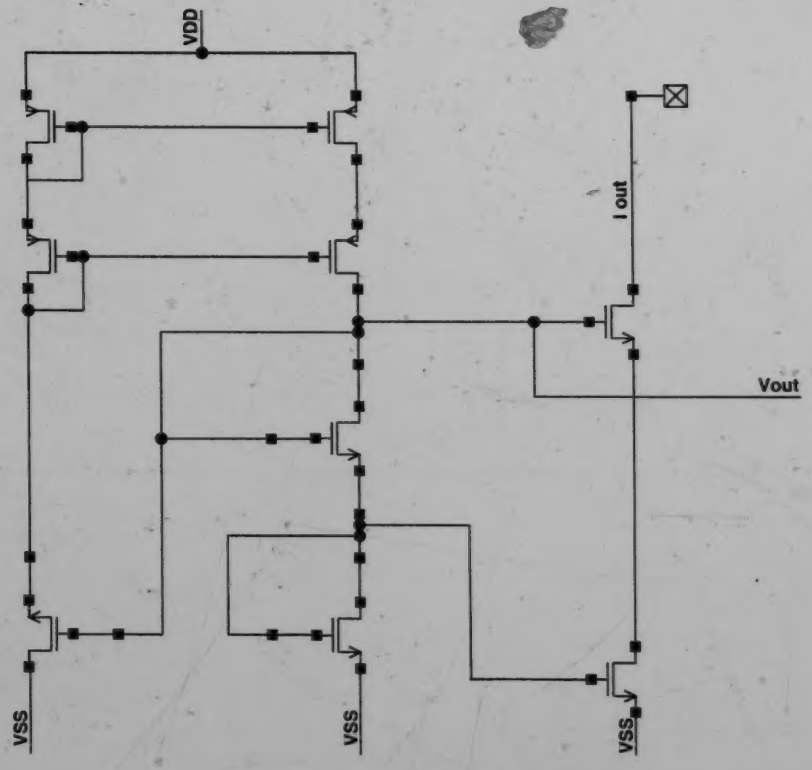


Figure 3.18: ML6 temperature probe

100 degrees Celsius. Simulation results showed a voltage of 500 mV corresponding to a temperature of 0 degrees Celsius and a voltage of 720 mV corresponding to a temperature of 100 degrees Celsius. There are two temperature probe points in the ML6 chip. One is situated in the periphery of the chip while the other is positioned closed to the memory core. Measurements from each of the temperature probe points can be read from output pins in the ML6 chip.

The actual on-chip temperature probe can be calibrated using a thermocouple (attached to the chip package) on a digital multimeter while connecting the temperature probe output to an oscilloscope for observation. The temperature probe can be calibrated by observing the voltage changes on the oscilloscope and the multimeter as the temperature is increased by an external source, from room temperature.

Chapter 4

ML6 Specifications and Chip Simulation

The design of ML6 was verified using analog and functional simulations. The memory core was designed and entered in schematic form and each schematic module was simulated and verified in the analog environment. A reduced memory core consisting of ten bitlines was also simulated in an analog environment.

The periphery is designed in Verilog and simulated in a digital environment using the Verilog-XL simulator tool from the Cadence IC design tool suite. Analog pad-to-pad simulations were also done for the chip. The vectors used in the pad-to-pad simulations were those that were used in the functional verification of the ML6 chip.

4.1 Design Overview and Specifications

ML6 is a variable-capacity MLDRAM chip like ML5. ML6 is able to operate in 2, 3, 4, 5 and 6-level modes which are selected by the choice of control waveforms. The 5-level mode was not implemented in ML5. The new mode corresponds to a capacity of 1.25 bits per cell.

The ability to operate in 2, 3, 4, 5 and 6-level modes was made possible by the switch matrix as explained in Section 3.2.2. Figure 4.1 below shows the reference voltages and the thermometer coding for six-level operation.

The bitlines are connected horizontally and vertically between the 5 sections

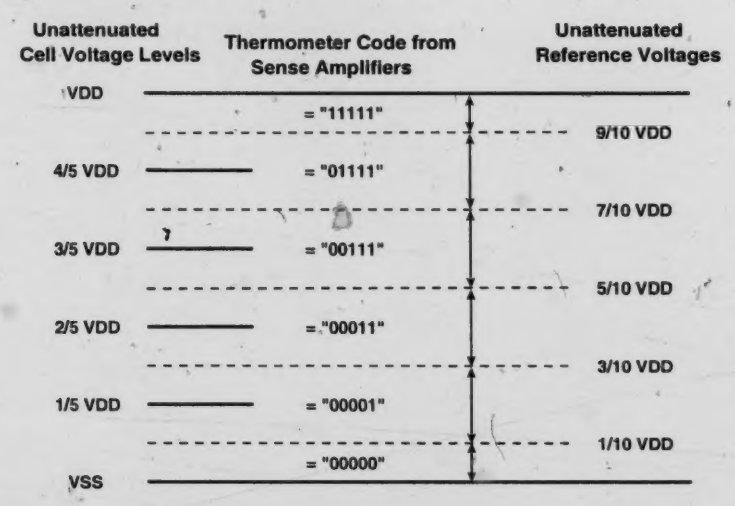


Figure 4.1: ML6 reference voltages and thermometer code for six-level operation

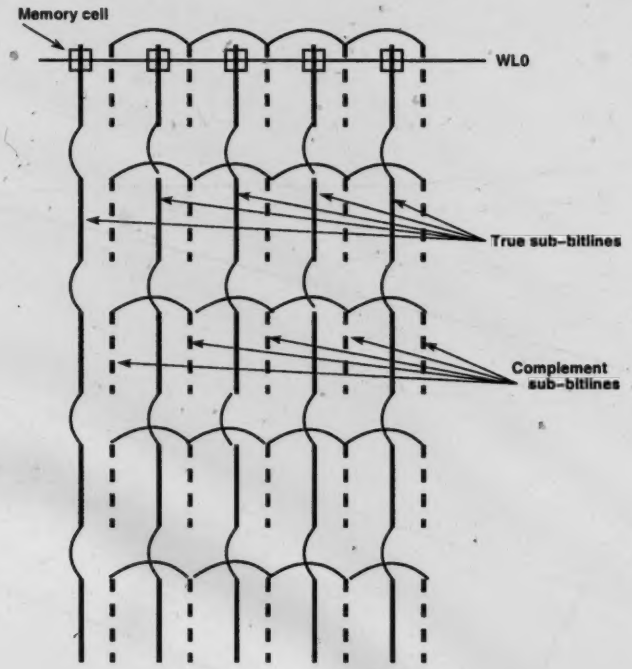


Figure 4.2: ML6 bitline connections during cell-dump before and after sensing

via switches controllable from outside of the chip. Figure 4.2, 4.3, 4.4 shows the positions of the switches in the chip. Charge sharing along the bitlines (vertically in Figure 4.2) is used to make multiple copies of the data cell signal. During reference

generation, charge sharing occurs (horizontally in Figure 4.2 among the bitlines to create the respective reference voltages in each of the 5 sections. For 6-level operation, 5 sections and bitlines are needed for charge sharing.

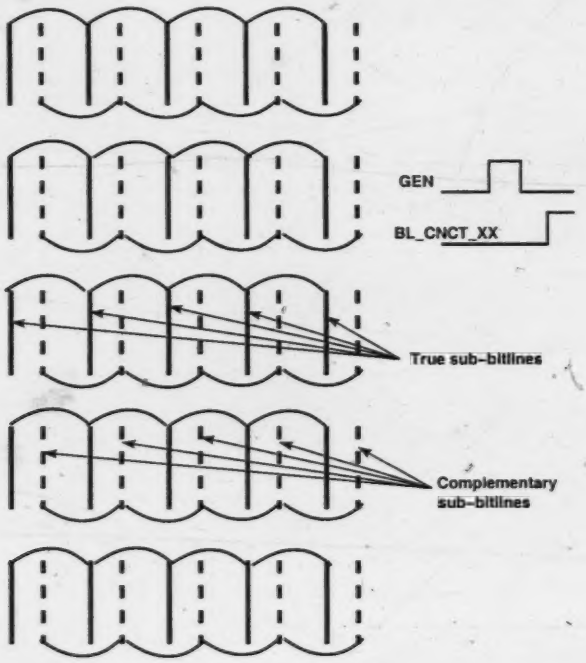


Figure 4.3: ML6 bitline connections during reference generation

For the other levels of operation, the switching mechanism is the same except that the switch matrix is reduced appropriately according to the reference voltages needed. As an example, for five-level operation, section A and the fifth bitline pair in the 5-by-5 array is not used. Therefore, the bitline-connect switches connecting section A to B, and the bitline-connect switches connecting bitline pairs 3 to 4 are kept off by the appropriate control waveforms during the sensing and reference generation operations.

4.2 The Testing Environment

The Agilent HP 81200 integrated circuit (IC) tester was used to test the ML6 prototype chips. The package type was PGA68 (68-pin pin grid array) [9], which has

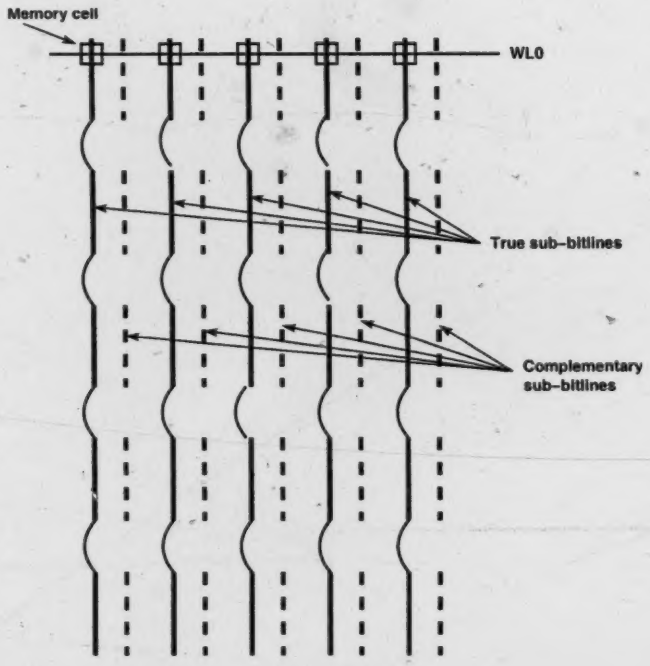


Figure 4.4: ML6 bitline connections during the restore operation

up to 68 pins. Sixty-four out of the possible 68 pins were used for the chip. In Appendix A, Table A.1 details the operating voltage and function of each functional and power pin on the chip. The table also shows how the die pads connect to the package pads. The pin number shown in the table corresponds to the pin number shown in Figure B.1(b) of Appendix B. The die diagram is shown in Appendix C.

Since the row and column addresses are descrambled before (after) the data is entered (or leaves) the core, the input and output data on the databus is seen as its true physical value ($V_{SS} = 0$ or $V_{DD} = 1$), as opposed to being flipped for odd memory cell access.

Table 4.1 details the power supplies used in ML6.

4.3 Chip Simulation

The pad-to-pad simulation of ML6 is discussed in this section. The six-level operation simulation considered since it uses the whole 5-by-5 array for sensing and

Table 4.1: Power Supplies in ML6

Power Supply	Description	Voltage
VDD_CORE	core power supply	1.8 V
VSS_CORE	ground	0 V tester reference potential
VDD_RING	ring power supply	3.3 V
VSS_RING	ground	0 V tester reference potential
VBB	memory array back-bias substrate voltage	-1.0 V
VBLP	bitline and sense amplifier precharge voltage	$1/2V_{DD}$
VPP	pumped voltage supply for the boosted wordline signals	2.0 V

reference generation. The simulation for the other levels of operation is similar to the six-level case except for the reduced number of bitlines and sections used in the operations. A reduced core is used in the simulations to greatly reduce the simulation time.

4.3.1 Analog Chip Simulation with Reduced Core

There are (320 BL × 12 WL) × 5 SECTIONS = 19200 addressable memory cells in ML6. The waveforms shown in this section correspond to writing to and reading from one example memory cell. The one memory cell from section A can be addressed at WLA[0] = 0 and colsel_A[0] = 0.

The following algorithm is used to test the memory cell functionality. The elaborate algorithm is used in the simulation since the vectors can be directly converted and used for the IC tester intended to test the chip.

1. Write a 0 to a column and row

2. Write a 1 to the column next to it on the same row

3. Write a 1 to the original column but on a different row
4. Read the data back from the original column and row
5. Verify read data against the expected data 0
6. Write a 1 to a column and row
7. Write a 0 to the column next to it on the same row
8. Write a 0 to the original column but on a different row
9. Then read the data back from the original column and row
10. Verify the read data against the expected data 1

The algorithm above is intended to check if there are unwanted coupling effects from the cells in one column to the cells in another column. Figure 4.5 illustrates the algorithm for writing and reading a 0 and then a 1 to a memory cell. Point 1 is the addressed cell. In the algorithm, say when a '0' is to be written into the cell, after writing the '0' value to the cell at point 1, the value of '1' is written into the cells at points 2 and 3 before reading back the value '0' from point 1. The algorithm is the same when a '1' is to be written into the cell at point 1. The cells are points 2 and 3 are also written with the value '1' as in the case for writing '0' to the cell at point 1.

For 6-level operation, the data is encoded as a 5-bit codeword as shown in Table 4.2.

The input data from DATA_IN is clocked in serially to each section of the array from the databus. At the same time, the same input values should appear on the output pin, DATA_OUT. In this way, we can verify if the data can be driven onto the databus from the input pin. The following details the sensing and reference generation operations for six-level operation. The waveforms on Figure 4.6 show the inputs going into the core for writing and reading of the data codeword '11111'.

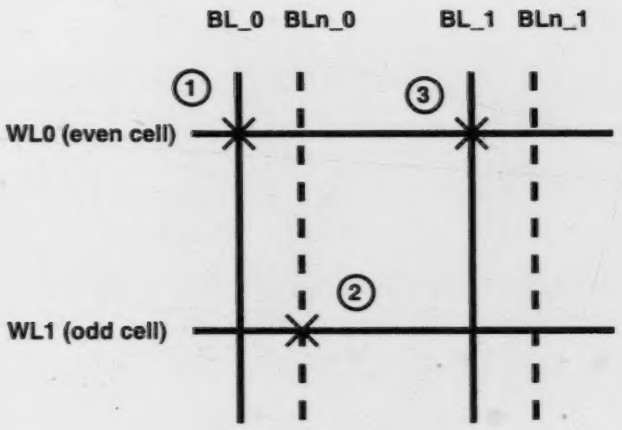


Figure 4.5: ML6 testing algorithm

Table 4.2: ML6 Six-level Operation Codeword and Voltage Representations

Five-bit codeword	Cell Voltage on True Bitline
00000	0
00001	$\frac{1}{5}V_{DD}$
00011	$\frac{2}{5}V_{DD}$
00111	$\frac{3}{5}V_{DD}$
01111	$\frac{4}{5}V_{DD}$
11111	V_{DD}

Write Operation At the start of the simulation, all bitline-connect switches are turned off and all sub-bitline sections are disconnected. The bitlines have been precharged to $\frac{1}{2}V_{DD}$. The ISO signal (time 0 ns in Figures 4.6 to 4.12 is high and this connects the sense amplifiers to the bitlines. Then, the row address is latched to permit accessing the row that contains the corresponding memory cell. XDEC_EN addresses the boosted wordline so that the value from the databus can be transferred onto the bitline for writing to the memory cell. At this time, one of the RGX signals goes high to turn on the appropriate reference and generate wordlines so that all the bitlines have the same number of memory (dummy) cells attached.

After the row address is latched, the corresponding column switch is ac-

tivated before the data at the input can be driven into the sections. The sense amplifiers are disabled after the write operation, and isolated from the databus leaving the bitlines floating at V_{DD} or V_{SS}. At the same time, XDEC_EN deactivates (time 70 ns) so that the write operation is completed when the voltage is trapped into the addressed memory cell.

Reference Generation Operation Before reading can be done, the five references have to be generated. RGX3 goes high to activate all reference and generate wordlines. The GEN signal is pulsed (time 80 ns) so that the sub-bitlines are precharged to one of V_{DD}, V_{SS} or $\frac{1}{2}V_{DD}$ values. The bitline-connect switches are then activated to connect the true and complement bitlines along all the columns so that charge sharing occurs and the appropriate reference voltage is created in each section. After charge-sharing, $0.1V_{DD}$, $0.3V_{DD}$, $0.5V_{DD}$, $0.7V_{DD}$ and $0.9V_{DD}$ are created in sections A, B, C, D and E, respectively. To capture the reference voltages in the reference sub-bitline, all RGX signals are deactivated. Only the reference cells in the reference sub-bitlines are used for the sensing operation.

Read Operation After reference generation, while the bitlines in the column are still connected, the remaining bitline-connect signals are used to connect the bitlines across the sections. Before the read operation, the sense amplifiers are again precharged to $\frac{1}{2}V_{DD}$. Next, the bitline-connect switches to the true bitlines across the sections, and complement bitlines along the columns, are deactivated. Again, this is done to minimize charge injection. At this point, the bitline precharge should already have stopped. The memory cell is then read by asserting the RGX2 signal, enabling XDEC_EN (time 170 ns) and deactivating the remaining bitline-connect switches. Pulsing YDEC_EN (time 200 ns) enables the data from each section to be read out onto the databus serially.

The restore operation occurs after sensing when the XDEC_EN addresses the appropriate row without pulsing the YDEC_EN and DB_SA_EN sig-

nals. The restore operation is similar to the write operation.

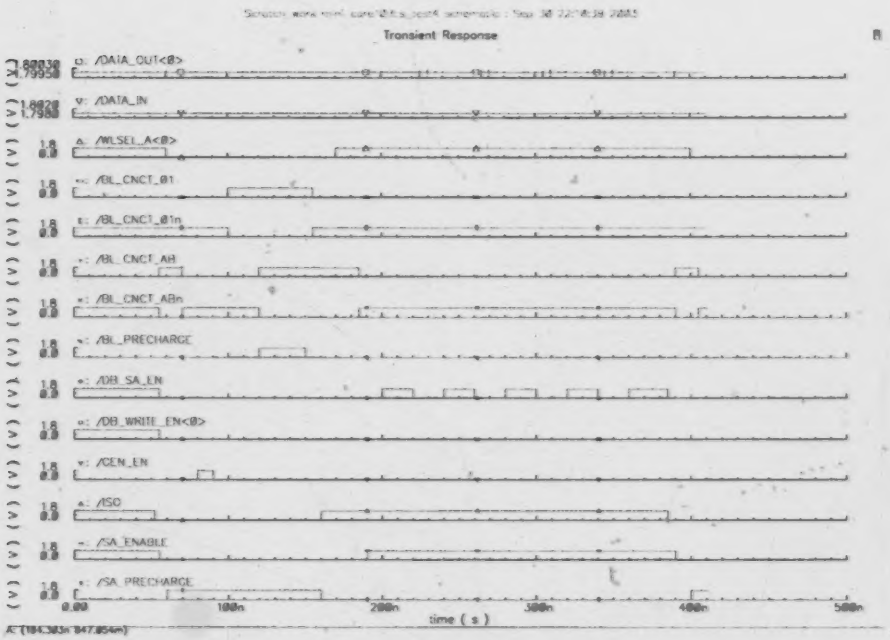


Figure 4.6: ML6 input waveforms going into the memory core

Simulation waveforms for the six possible input codes, 00000, 00001, 00011, 00111, 01111 and 11111 are also shown in Figures 4.7 to 4.12.

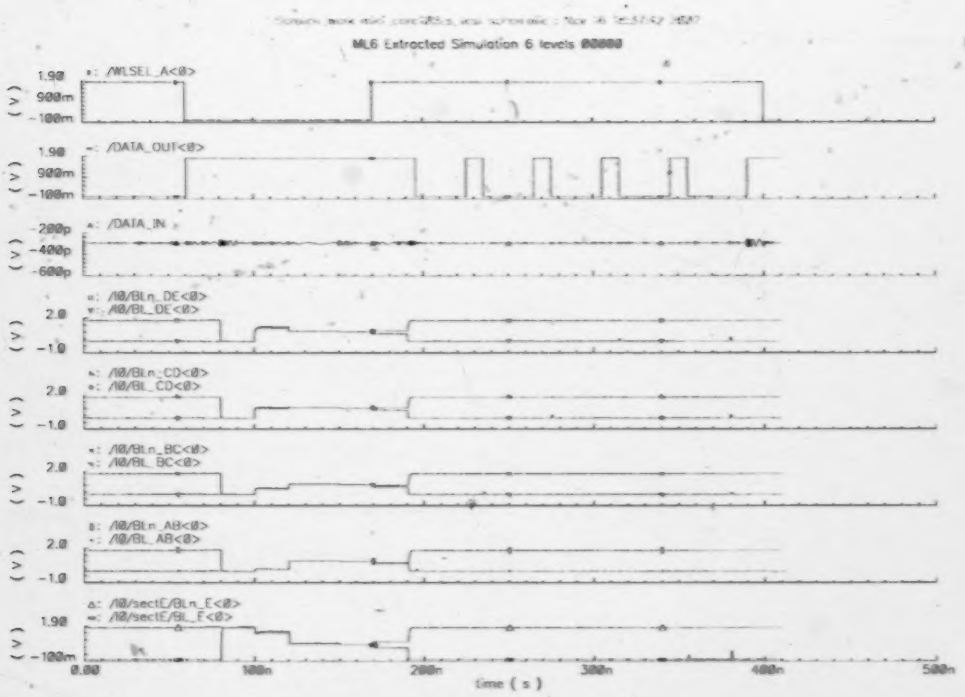


Figure 4.7: ML6 output and bitline waveforms for input = '00000'

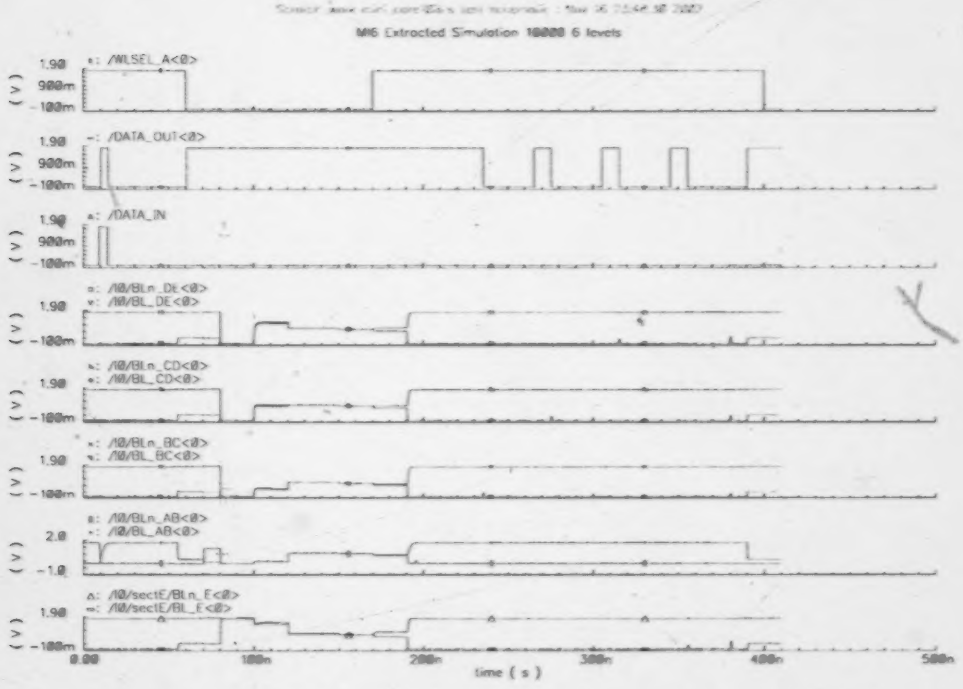


Figure 4.8: ML6 output and bitline waveforms for input = '10000'

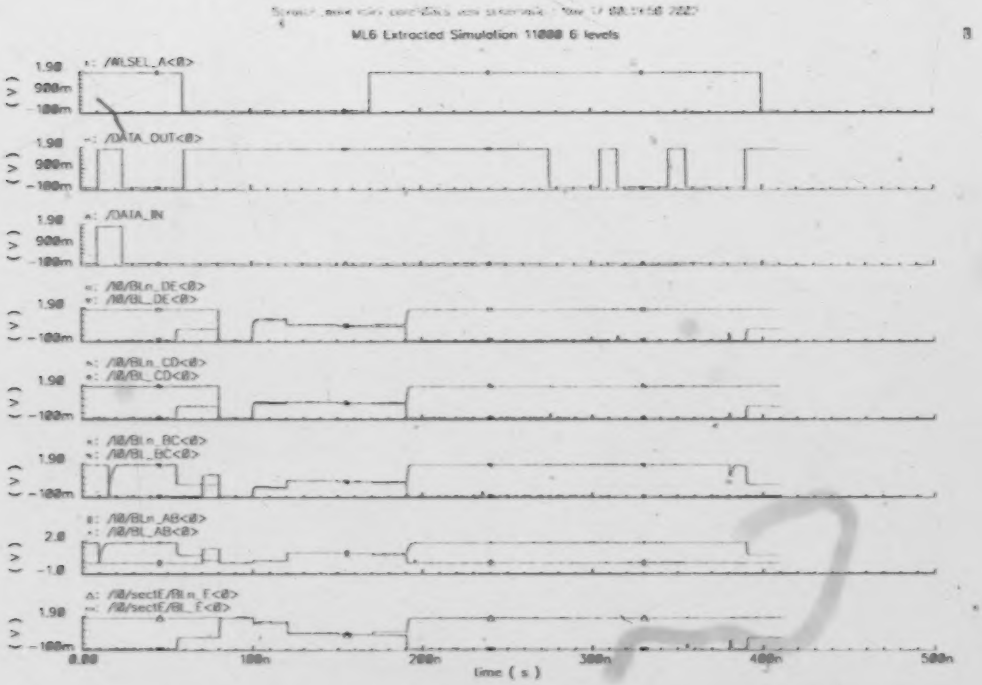


Figure 4.9: ML6 output and bitline waveforms for input = '11000'

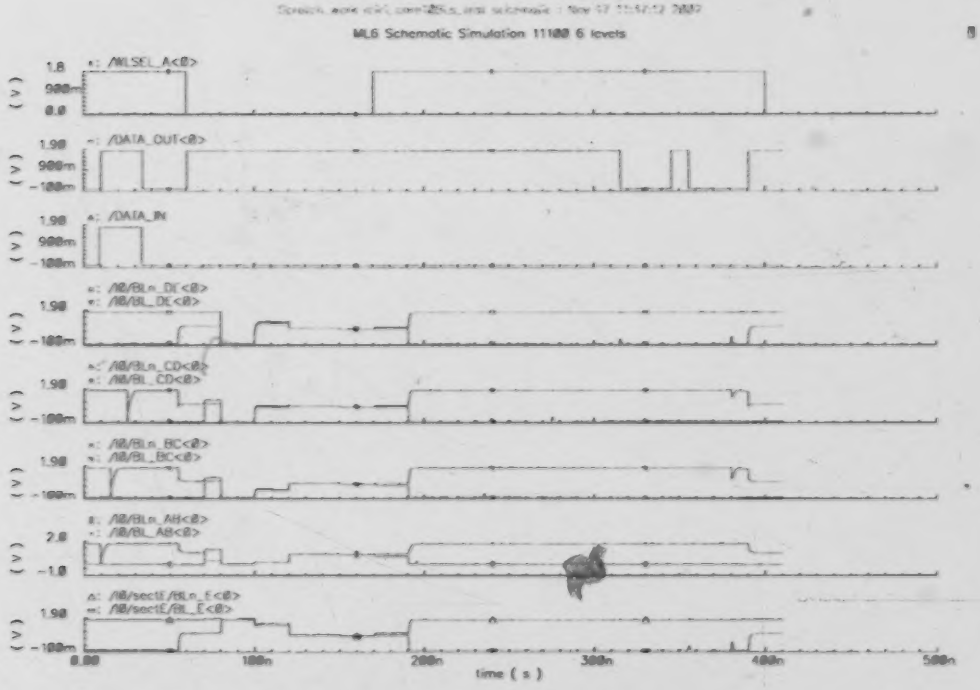


Figure 4.10: ML6 output and bitline waveforms for input = '11100'

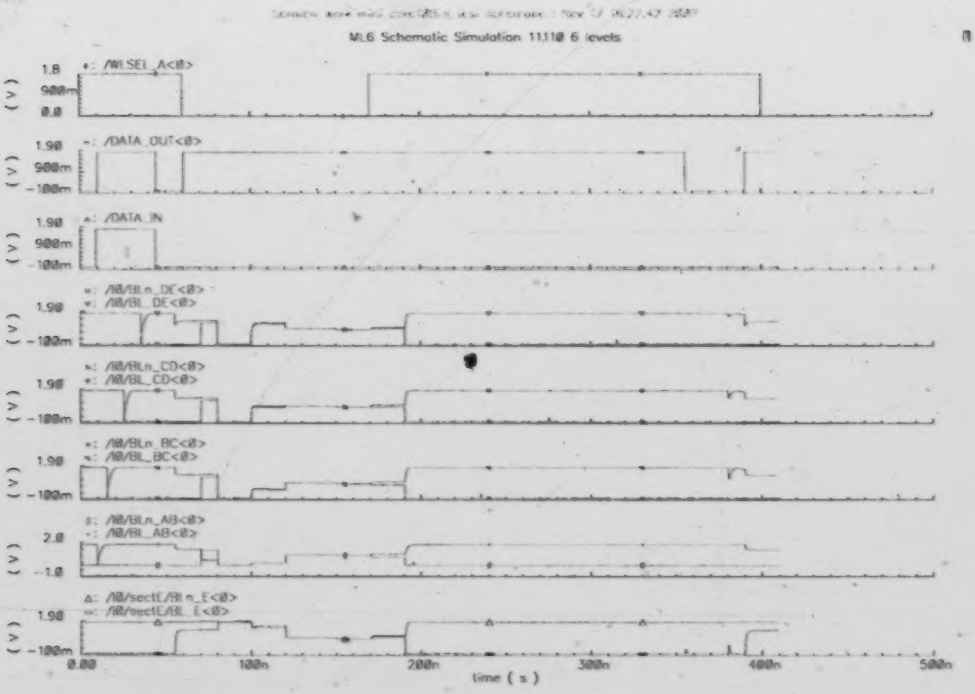


Figure 4.11: ML6 output and bitline waveforms for input = '11110'

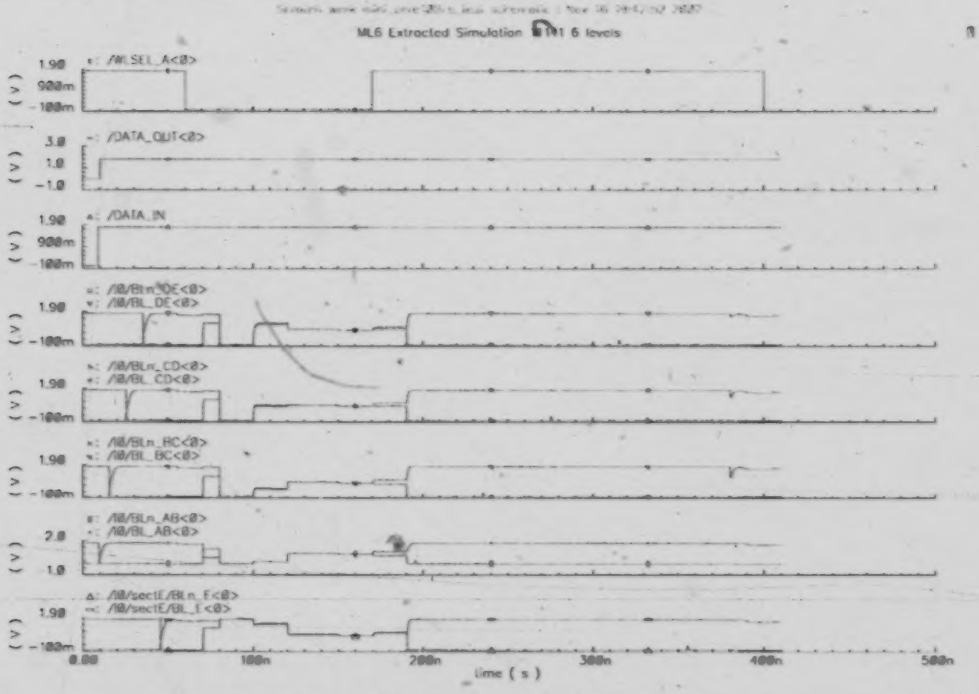


Figure 4.12: ML6 output and bitline waveforms for input = '11111'

Chapter 5

Test Chip Debug and Evaluation

The ML6 chip was fabricated using the 0.18-μm mixed signal process technology from TSMC. Cadence design tools, and TSMC's Design Rule Checker (DRC) and Layout-versus-Schematic (LVS) rules were used to enter and verify the design. The ML6 chip is packaged in the PGA68 package available from CMC [9] with a total pin count of 42 inputs and 2 outputs, as described in Chapter 4. The maximum number of pins in this package is 68.

A total of 10 packaged dies and 40 loose dies were received from the CMC. Of the 10 packaged dies, two have been tested. The results of the functional tests are presented and discussed in this chapter.

An Agilent HP 81200 VLSI tester supplied by CMC was used for testing ML6. The test vectors were generated using a C-program previously written for ML5. The C-program was modified to work for ML6.

One chip was selected for debug and initial testing. After the debugging the functional test program, the first prototype chip was tested at all levels of operation. One other packaged chip was tested. The eight remaining packaged die were not tested due to time constraint and to preserve them for further work by other students.

To simplify testing of the full chip, a Visual C++ tester interface program used in ML5 was modified for ML6. The program controls the tester through a Graphical User Interface (GUI) platform when loading the appropriate vectors and testing all memory cells in a chip. With this GUI interface there is no need to use the Agilent user software, which is less suited for data collection from multiple cells. The

Visual C++ GUI program is able to collect, interpret, sort and finally output the data into a readable text format.

New PERL scripts and a C-language program previously used in ML5 were used to manipulate the output data. The output data were also presented in the form of coloured bitmaps to more clearly show any patterns or alignments of failing cells. The bitmap generation programs are modifications of the ones used in ML5.

5.1 Debug History

Debugging of the test program along with the first chip started with the testing of databus functionality, followed by sense amplifier functionality and cell read or write functionality. Next, simple DRAM (two-level) operation was tested. Finally, the more complicated three-level, four-level, five-level and six-level operating modes were verified to complete the initial test of basic boolean functionality for ML6.

5.1.1 Databus Functionality

There are 40 databus pairs in ML6, and every databus pair can be connected to one of eight sense amplifiers and bitlines. The databuses are precharged to V_{DD} before any sensing is done. The databus precharge is set when the Y address decode enable (Y_DEC_EN) signal is deasserted. There is a built-in circuit delay[1] of 2 ns before the databus precharge is activated after the deassertion of the Y address decode signal. Each databus pair is built with its own differential sense amplifier for sensing on the databus. Figure 5.1 shows the connections and locations of the databus and bitline circuits.

The first check of the databus was to see if the precharged databus could be pulled down low after being precharged to V_{DD}, and then pulled up high again from V_{SS}. This checks if the databus is stuck-at any value. A simple test would be to write to one databus the values '1', '0' and then '1'. During this initial check, the

[1]The delay was implemented in the one-shot circuit by inserting delay inverters to make up the 2 ns delay. The circuit delay was simulated in the analog environment.

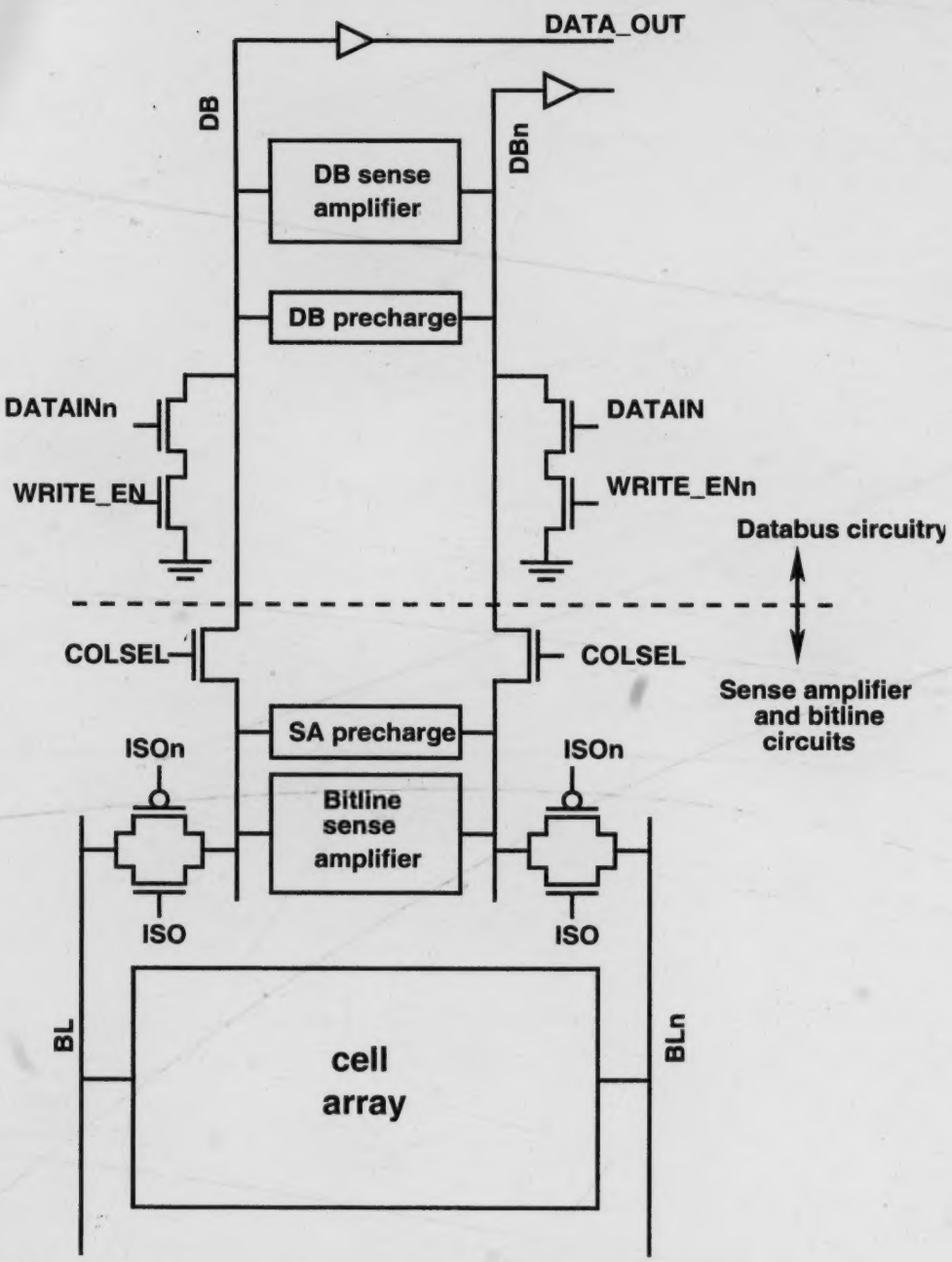


Figure 5.1: Databus and bitline circuitry

sense amplifiers are turned off and the wordlines are not used. The sense amplifiers are also separated from the bitlines by deasserting the isolation switch connecting

the sense amplifiers to the bitlines (ISO).

Figures 5.2 and 5.3 shows the input waveforms for the databus check. The WRITE_EN and the YDEC_EN signals toggle at the same time. The DB_SA_EN signal is also activated to drive the data put onto the databus to the correct value. The output of the chip (DATA_OUT) shows that the databus can be driven to V_{DD} and V_{SS} and is not stuck at any one value.

All forty of the databuses were checked in the same way as described above by changing the Y address to the chip and repeating the test.

5.1.2 Sense Amplifier Functionality

There are as many sense amplifiers in ML6 as there are bitlines in every section. All of the 320 bitline pairs are connected to a sense amplifier in each section, giving $(320 BLs) \times (5 sections) = 1600$ sense amplifiers per chip. The bitline sense amplifier is a differential latch amplifier commonly used in DRAM designs.

Checking the sense amplifier is difficult. While there is isolation from the bitlines connected to the memory cells, the sense amplifier is connected to the databus by the column-select switches that are directly controlled by YDEC_EN. When YDEC_EN is deasserted, the databuses are being precharged to V_{DD}. In this way, we cannot write to the databus independently first before selecting the appropriate bitline (and hence the sense amplifier) to write onto (by asserting the YDEC_EN signal). On top of this, the databus and bitline sense amplifiers may "fight" against each other momentarily due to glitches on the bitlines caused by untimely switching of signals.

Therefore, the sequence of the sense amplifier check should be to write to the sense amplifier and the databus at the same time by asserting YDEC_EN and WRITE_EN on the databus — without activating either the databus or bitline sense amplifier. The bitline sense amplifier can then be activated after the YDEC_EN and WRITE_EN signals are deasserted. This allows the databuses to be precharged and the bitline sense amplifier to be isolated from the databus while it amplifies the data voltage to the correct value.

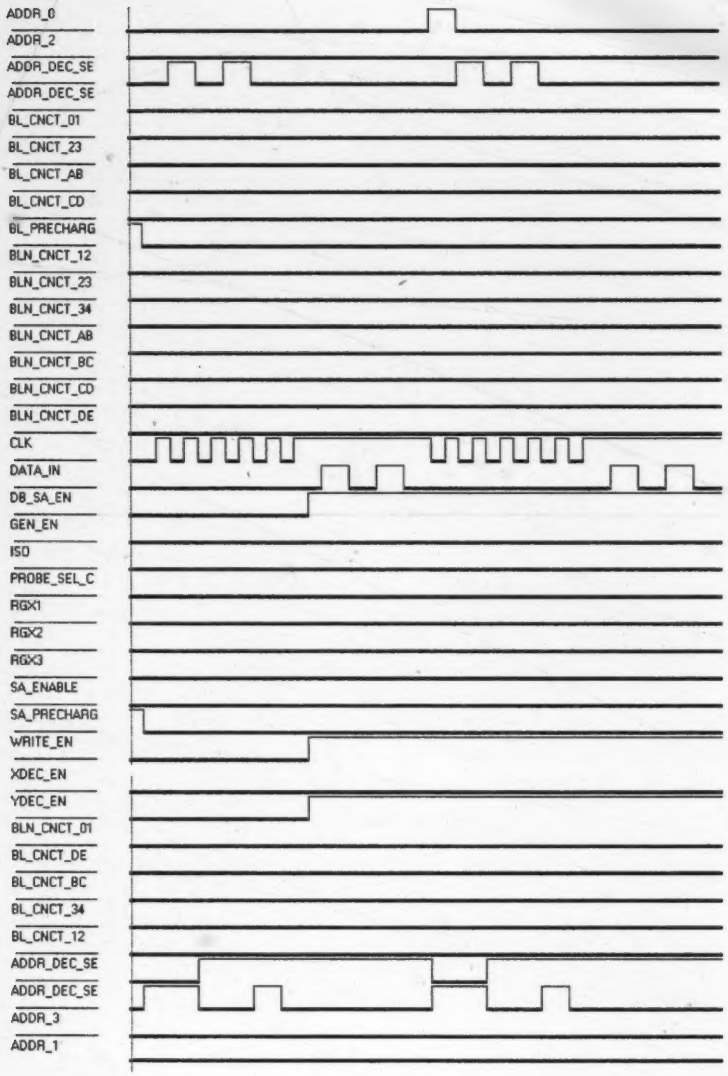


Figure 5.2: ML6 databus debug input waveforms

To read back from this bitline sense amplifier, it is first switched off. The bitline sense amplifiers can be switched off at this time since the correct data values would already have been amplified and the bitlines would contain the correct data voltage values. Then the YDEC_EN signal can be asserted to select the bitline so that the data value from the bitline sense amplifier can be transferred to the databus. While the YDEC_EN signal is kept asserted, the databus sense amplifier can be activated

July 11_03_db/output_db

BL SA is disabled.
Only databus SA is enabled.
Toggling data_in writes 1-0 to the databus.

This experiment works for even and odd WL addressing.

Conclusion:
The write circuitry is able to drive the databus to any input values.
The databus SA is able to drive the databus to the write value.

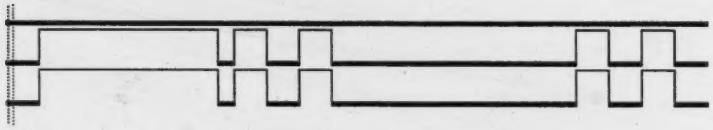


Figure 5.3: ML6 databus debug output waveforms

so that the correct logic value can be amplified. Figure 5.4 shows this sequence in the input signals. Figure 5.5 shows the outputs from various experiments done.

5.1.3 Memory Cell Functionality

The cell functionality check is an extension of the sense amplifier experiment previously described. The databus and sense amplifier enable input waveform sequences are similar. To access the memory cells, the ISO and XDEC_EN signals have to be asserted so that the bitline sense amplifiers are connected to the bitlines and the appropriate wordline can be selected for cell access. Figures 5.6 and 5.7 show the input and output waveforms for the memory cell verification experiment.

It is found during the debugging stage that the tester is unable to delete and initialize the contents of the memory cell before writing to the cell. Powering the tester down and then up again does not clear the contents of the memory cells. As a result, the old data value in the cell may interfere with the sensing of the new data value. To solve this problem, the memory cells can be first "cleared" of the previous data value by leaving the bitline precharge on for a split second when the wordline is asserted for sensing. Figure 5.8 shows the waveforms that can be used to perform this task.

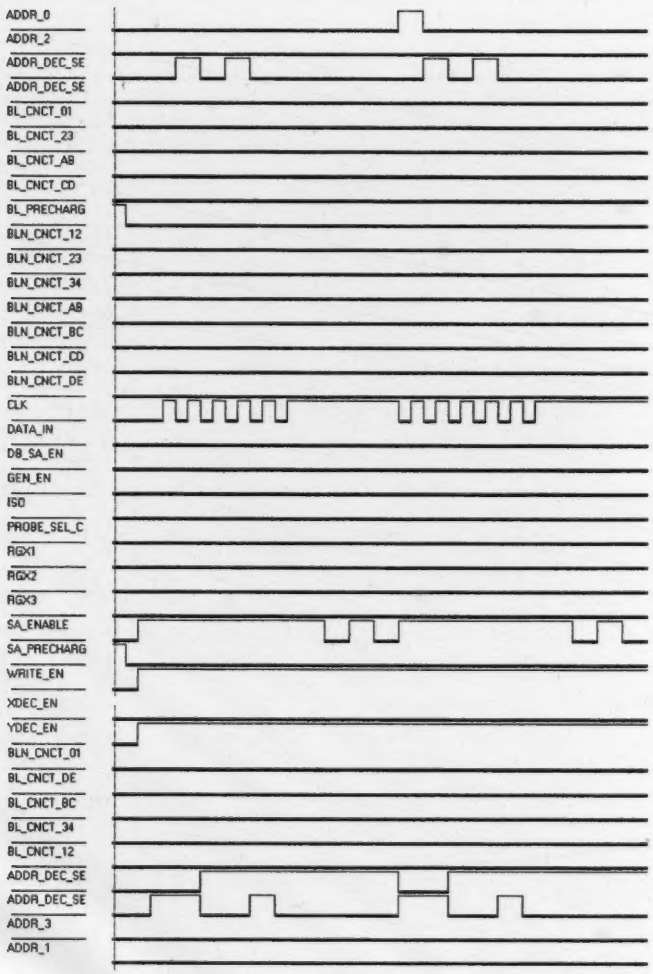
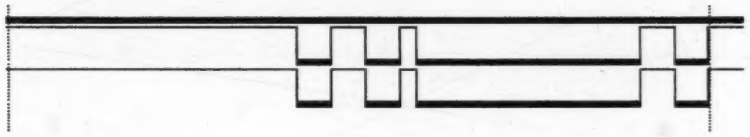


Figure 5.4: ML6 sense amplifier debug input waveforms

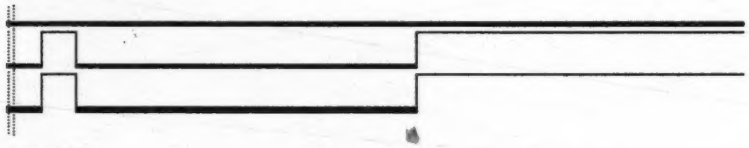
Experiment 1

write_en is turned off
db_sa_en is toggled at various data_in values to show what happens when write_en is off.
db_sa_en is used to enhanced the signal changes.



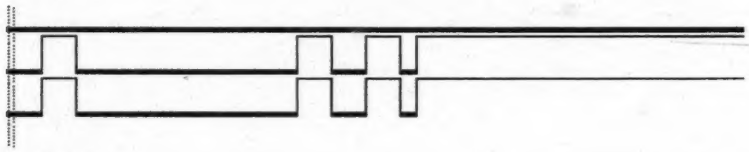
Experiment 2

db_sa_en is turned off
write_en is turned on
sa_enable is turned on
data_in is toggled at even and odd WLs



Experiment 3

Same signal environment as Experiment 2 except that data_in is kept at 1 while sa_enable toggles.



Experiment 4

Same signal environment as Experiment 2 except that data_in is kept at 0 while sa_enable toggles

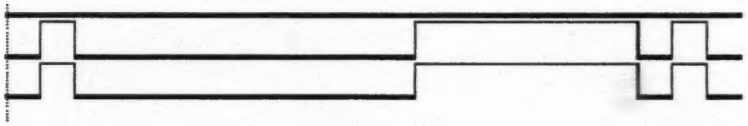


Figure 5.5: ML6 sense amplifier debug output waveforms

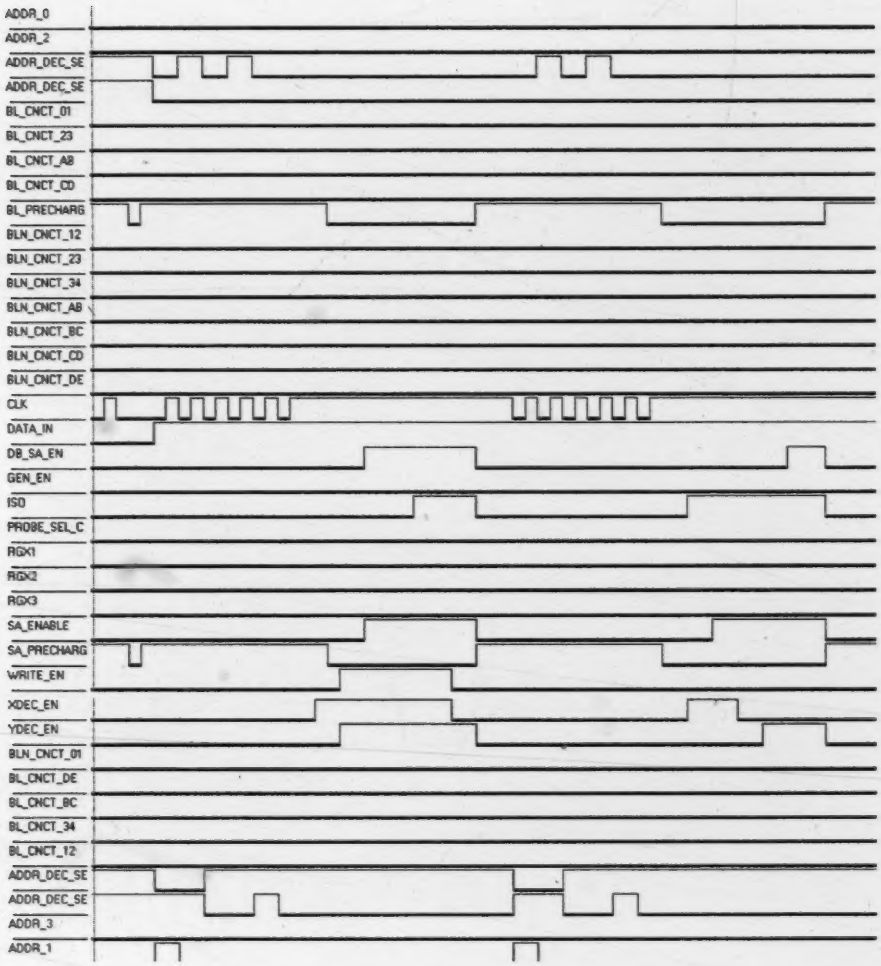


Figure 5.6: ML6 cell debug input waveforms

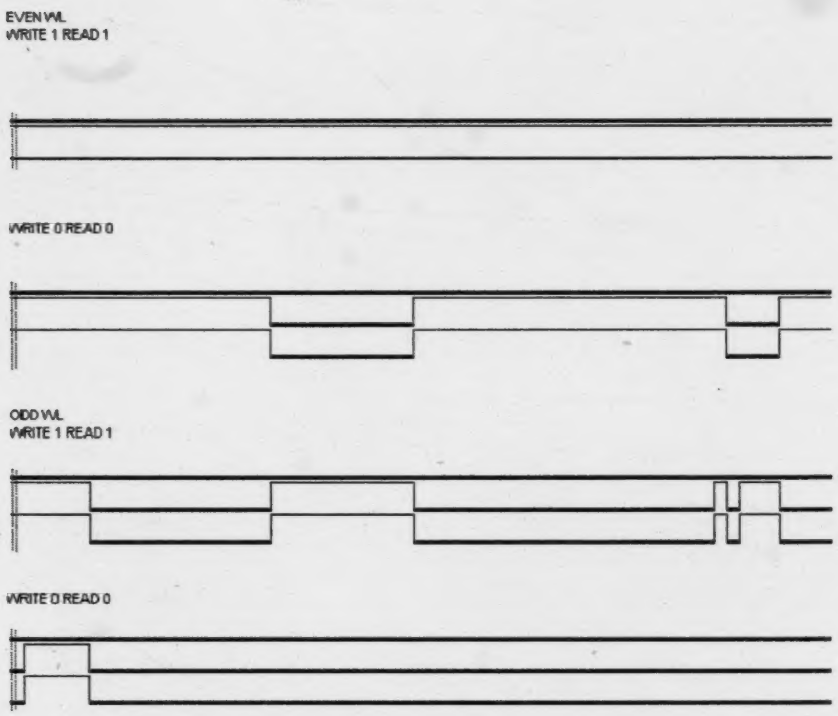


Figure 5.7: ML6 cell debug output waveforms

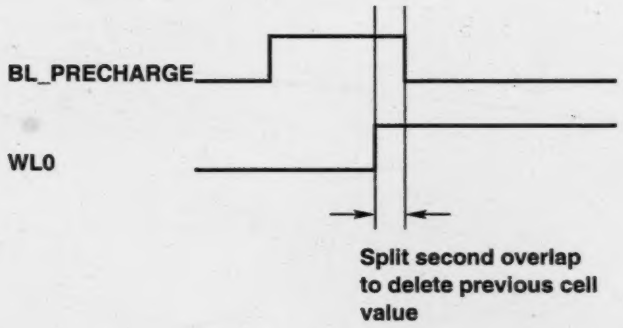


Figure 5.8: Waveforms used to 'delete' the content of memory cells before a write operation

5.1.4 Two-level (DRAM) Operation

The RGX1, RGX2 and RGX3 control signals are not used for two-level (DRAM) operation as they are only used to access the reference and generate cells, and for bitline capacitance balancing during MLDRAM sensing. Two-level operation is simpler in that the $\frac{1}{2}V_{DD}$ sense amplifier and bitline precharge value can be used as the reference value during sensing (writing or reading). This avoids the use of the RGX1, RGX2 and RGX3 signals and the complicated sense and restore operations required in the multilevel modes.

The sensing sequences are similar to the ones used for the memory cell functionality check with the additional use of the bitline connect switches in between the bitlines and sections. In two-level DRAM operation, asserting the bitline connect switches in between the sections make the bitlines longer so that the bitline to cell capacitance ratio during read and write operations are closer in magnitude (each bitline segment is about 11 fF while the cell capacitance is about 40 fF). Connecting the bitlines along the sections will make the capacitance of the bitline five times as large. This helps in making the read and write operation more stable and repeatable. The switches are kept on the whole time to minimize switching noise (on during the actual read and write operations). Keeping the switches off, the tester was found to sometimes read different output values on tests on the same memory cell.

Figures 5.9 and 5.10 show the input and output waveforms for 2-level, DRAM operation, on ML6. Both the read and write operations follow the following sequence of events: precharge (reference generation), set X address, cell dump, set Y address, sense (write or read), restore. This sequence of operation which includes the cell dump and restore operations before and after the actual sensing (writing or reading) operation allows multiple cells to be written before they are read out again. This is because the read is not destructive. The contents of the cells are restored after the sensing operation. The cell dump operation puts the contents (i.e., shares the charge) of the addressed cell onto the bitlines for reading. During the write operation, the cell dump operation still puts the values from the addressed cell onto

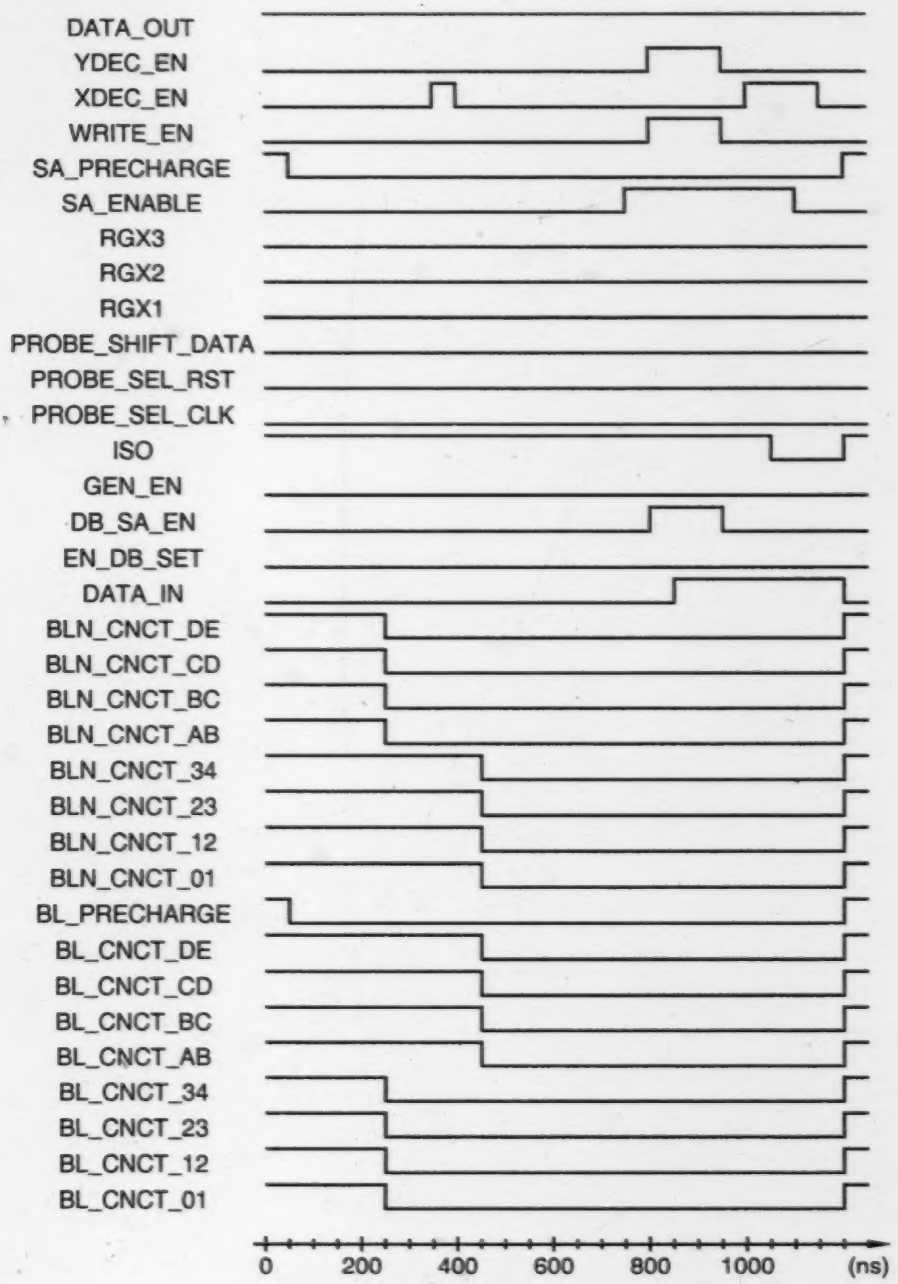


Figure 5.9: ML6 two-level operation write waveforms

the bitlines, but the contents of the cell will be over-written after the sensing step during the restore operation.

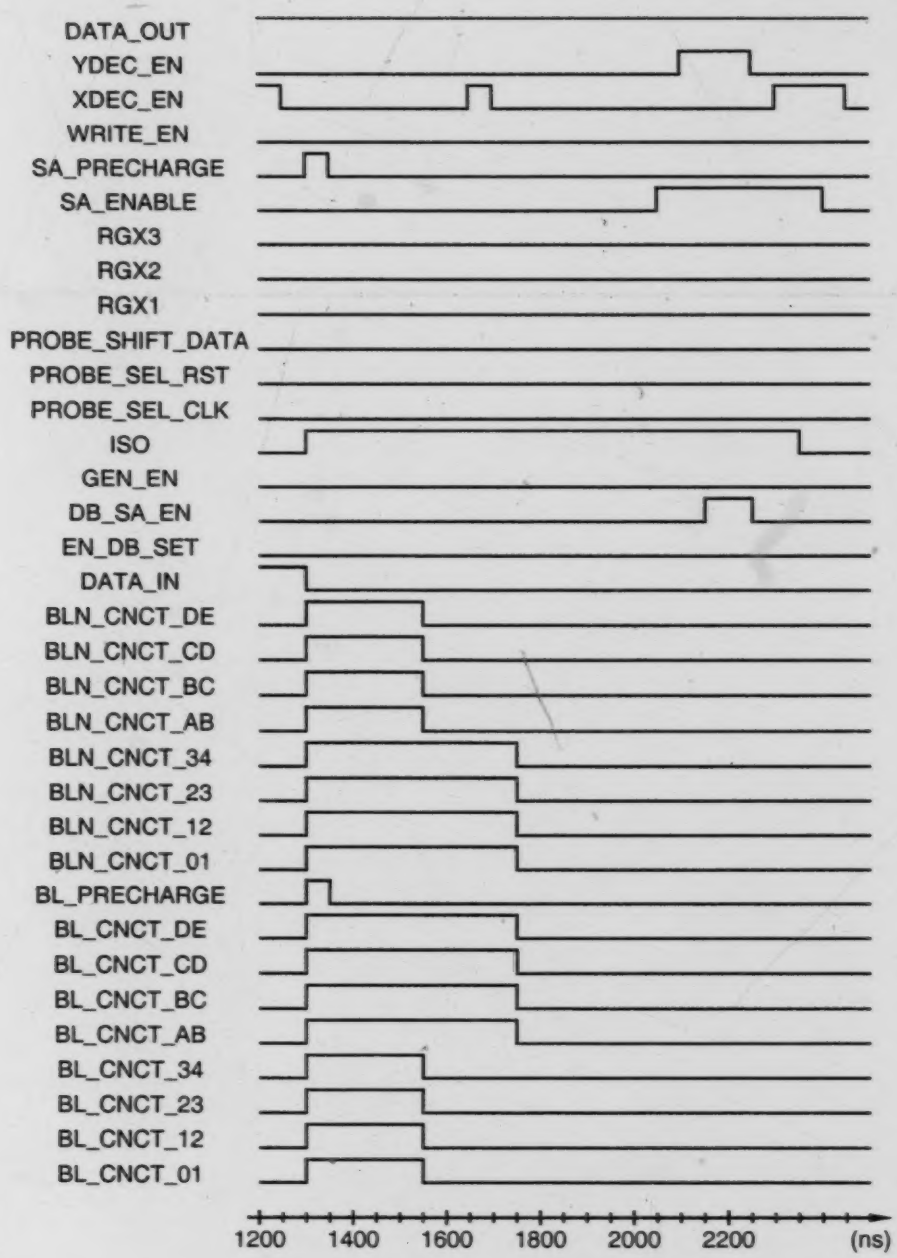


Figure 5.10: ML6 two-level operation read waveforms

1. **Precharge** The whole cycle of writing and reading starts with precharging the databuses, sense amplifiers and bitlines (time 50ns in Figure 5.9). At this point, all the other input signals are set to '0' and the databus

(DATA_OUT) produces a '1' due to YDEC_EN being set to '0' and the databus precharge to V_{DD}. Since the precharge value on the bitlines and sense amplifier is $\frac{1}{2}V_{DD}$, it can be used as the reference voltage and there is no need for a separate reference generation operation.

2. **Cell Dump** Although the cell dump operation is present in both the read and write cycles, it is only used during the read cycle where the contents of the cells are read during sensing. In the write cycle, the contents of the cell will be written over during sensing.

3. **Sense — Write or Read** During the write cycle, the following signals are used to put a value on the databus, bitline and sense amplifier: YDEC_EN, WRITE_EN, DB_SA_EN, SA_ENABLE and ISO. Asserting WRITE_EN (time 800 ns in Figure 5.9 puts a value onto the databus while YDEC_EN transfers the value to the addressed bitline and sense amplifier. After the writing operation, the value on the bitline is written and captured into an addressed cell by asserting and then deasserting XDEC_EN which selects the appropriate wordline.

   The same control signals are used for reading except for the WRITE_EN signal, which is kept deasserted. Asserting the YDEC_EN signal (time 800 ns in Figure 5.9 will access the bitline which has the correct data value. The correct data value would already have been on the bitline resulting from the cell dumping operation before sensing. This value will be transferred to the databus and read out onto the DATA_OUT signal. The read operation is non-destructive as the values are restored to the cell after the sensing by asserting and deasserting the XDEC_EN signal again.

4. **Restore** The restore operation happens after sensing. For the write cycle, the restore operation, puts the data from the bitlines to the addressed cell. For the read cycle, the restore operation restores the original value to the cell by writing to the addressed cell, as in the write cycle.

5.1.5 Three-level Operation

For three-level operation, the same sequence of operations is used as in two levels, with the addition of a reference generation stage before the cell dump. This is because the reference now must be generated before sensing, unlike in two-level operation where the reference voltage can be the same as the precharge value, $\frac{1}{2}V_{DD}$.

The two reference values needed for a three-level operation MLDRAM are $\frac{1}{4}V_{DD}$ and $\frac{3}{4}V_{DD}$. These two reference values are generated by manipulating the switch matrix circuits in between the sections of the memory arrays in the chip. The switch matrix is built in the form of a 5-by-5 array. For three-level operation, the switch matrix is manipulated by asserting the appropriate switches to form a 2-by-2 array.

The following sequence of operations is used for writing and reading in three-level mode: precharge, reference generation, set X address, cell dump, set Y address, sense (write or read) and restore.

Similar to the two-level operation described previously, the databuses, sense amplifiers and bitlines are first precharged while the other input signals remain deasserted.

During reference generation, the sections and horizontal bitlines are first isolated from one another by deasserting the appropriate bitline-connect switches. Next, the appropriate voltage values are driven onto the bitlines by asserting the GEN_EN signal. At this point the bitlines will contain the voltage values ready to be charge-shared so that the correct reference voltages can be created. As mentioned before, there are two ways to generate the reference voltages. Any of the two combinations of bitline connections will generate the appropriate reference voltages for three-level operation. During the whole reference generation process, the RGX1, RGX2 and RGX3 signals are asserted so that the generated reference voltages will be written to the reference and generate cells. Only the values on the reference cells are used for sensing. The generate cells are used for balancing the capacitance of the bitlines during reference generation, cell dump and restore operations to ensure

more accurate sensing.[2]

The cell dump and restore operations in three-level mode work the same way as for the two-level mode. The write or read operations, however, are more complicated as multiple sections on the same bitline have to be accessed to put the appropriate values onto the bitlines to be charge-shared. For three-level operation, the thermometer code for a two-bit pair is written onto the appropriate sections. After writing the values onto the bitlines, the bitlines in between the sections are connected to create the correct value for storing in one of the cells. For the case of using sections B and C, across bitlines 1 and 2, the control signal YDEC_EN has to be toggled twice so that one of the following bit pairs can be written onto the correct sections. To put the appropriate values into the memory cell during the restore operation, BL_CNCT_BC is asserted while the remaining bitline-connect switches are turned off. At the same time, RGX1 and RGX2 are asserted to turn on the reference and generate wordlines in sections other than the section containing the addressed wordline, and the reference cells in all sections The valid thermometer codes for three-level operation are 00, 01 and 11.

Before sensing, the cell dump operation shares the charge stored in the addressed cell onto the bitlines. At this time, the bitlines in between the sections should be disconnected and the reference cells in each section should contain the correct reference voltage values. During the cell dump operation, the RGX2 signal is asserted so that the reference values in each section are transferred to the respective bitline sections. At this time, the bitline switches in between the sections should be off so that the sections are separated from one another. This analog voltage is then compared to the reference voltages in the relevant sections. For the case of three-level operation, the relevant reference voltages would be in sections B and C, or sections C and D.

During the sensing operation, YDEC_EN is toggled as the relevant sections are addressed. For reading from the cell, toggling the YDEC_EN signal accesses the

[2]Section 3.2.2 of Chapter 3 explains switch matrix and reference-generate mechanism in more detail.

data voltage values from each section along the addressed bitline and puts the amplified data onto the databus. For writing into a cell, toggling the YDEC_EN signal puts switches the digital data voltage from the databus into each section along the bitline to be diluted and kept in a memory cell when XDEC_EN is toggled (restore operation) before the actual sensing occurs.

Figures 5.11 and 5.12 shows the input and output waveforms for three-level operation.

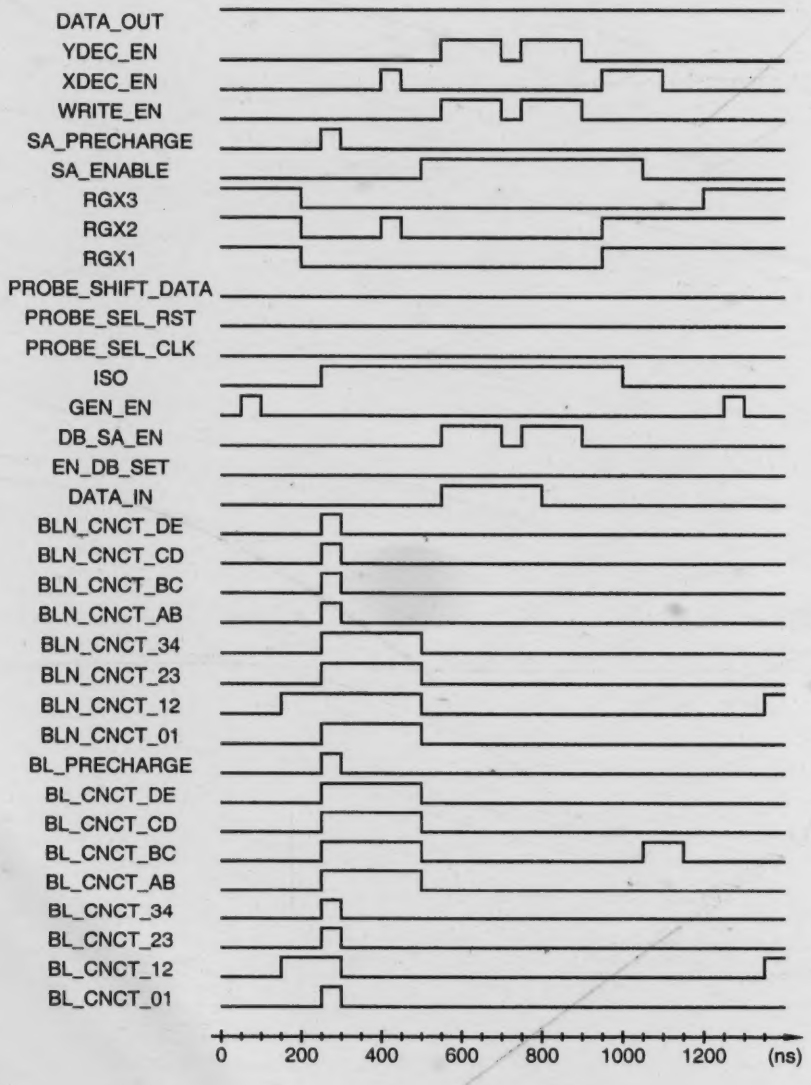


Figure 5.11: ML6 three-level operation write waveforms

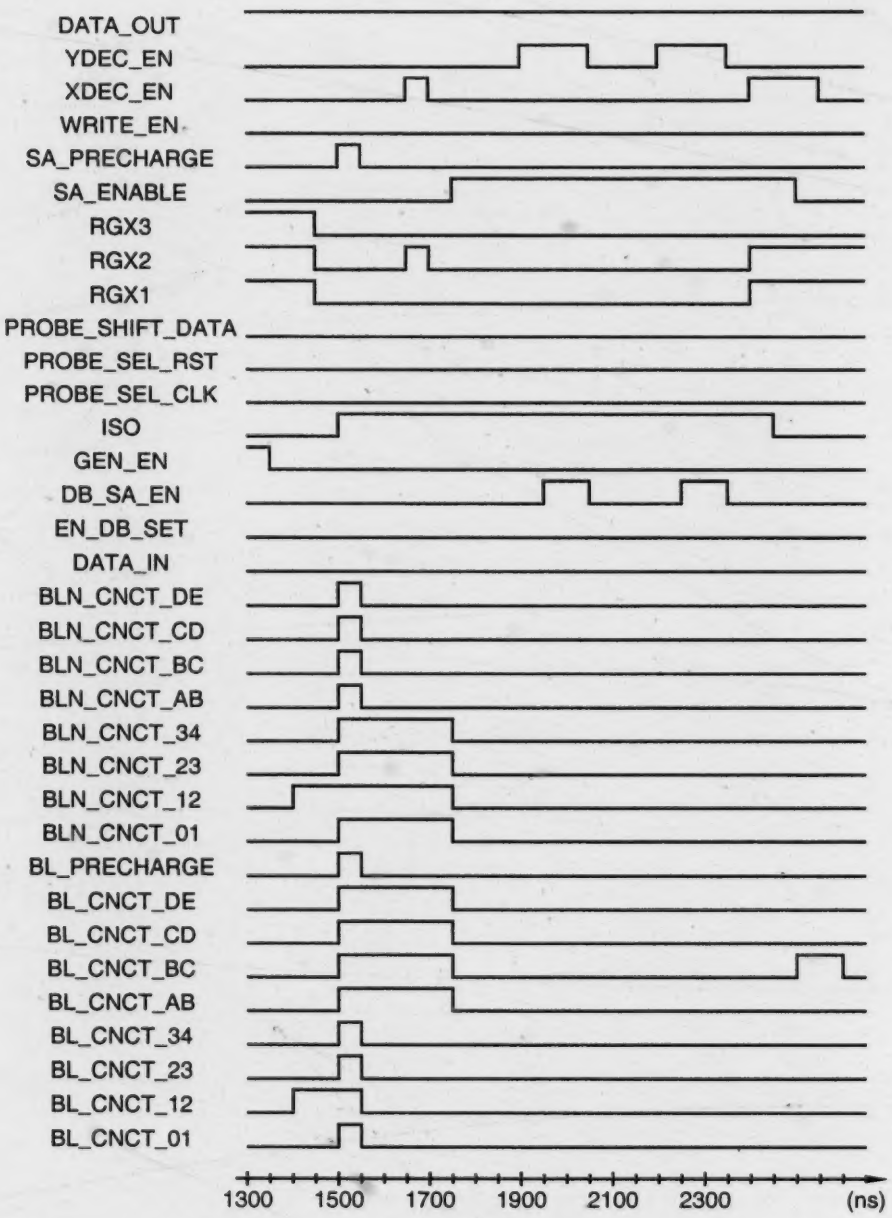


Figure 5.12: ML6 three-level operation read waveforms

5.1.6 Four-level Operation

The input and output waveforms for four-level operation are similar to those for the three-level mode except for differences in the reference generation switch matrix combination and the sensing operation.

The switch matrix can be used in combination of the voltages to create the reference values for four-level operation. With the type of voltage values used in the matrix, there is only one combination of voltages that can be use for the reference voltage generation for four-level operation. The reference voltages for four-level operation are $\frac{1}{6}V_{DD}$, $\frac{3}{6}V_{DD}$ and $\frac{5}{6}V_{DD}$. These voltages can be computed from the combination of voltages from sections B, C and D across from bitlines 1, 2 and 3.

The sensing operation for four-level operation involves accessing three different sections and expecting to sense one of the four 3-bit wide valid codes as follows: 000, 001, 011 and 111. Accessing the three different sections sequentially means toggling the YDEC_EN signal three times in a row during sensing. Once the values are put into the appropriate sections, the bitlines in between the sections are connected so that the charge sharing will dilute the combined voltage over the full-length bitline. The sense amplifier then amplifies the values on the bitlines from rail-to-rail before the value is stored into an addressed memory cell during the restore operation. To read from the cell, the cell dump operation puts the analog voltage value from the addressed onto the bitlines. The sense operation after that will then compare the analog voltage from the different sections with their respective reference values as YDEC_EN toggles when the different sections are addressed.

Figures 5.13 and 5.14 show the input and output waveforms for four-level operation on ML6.

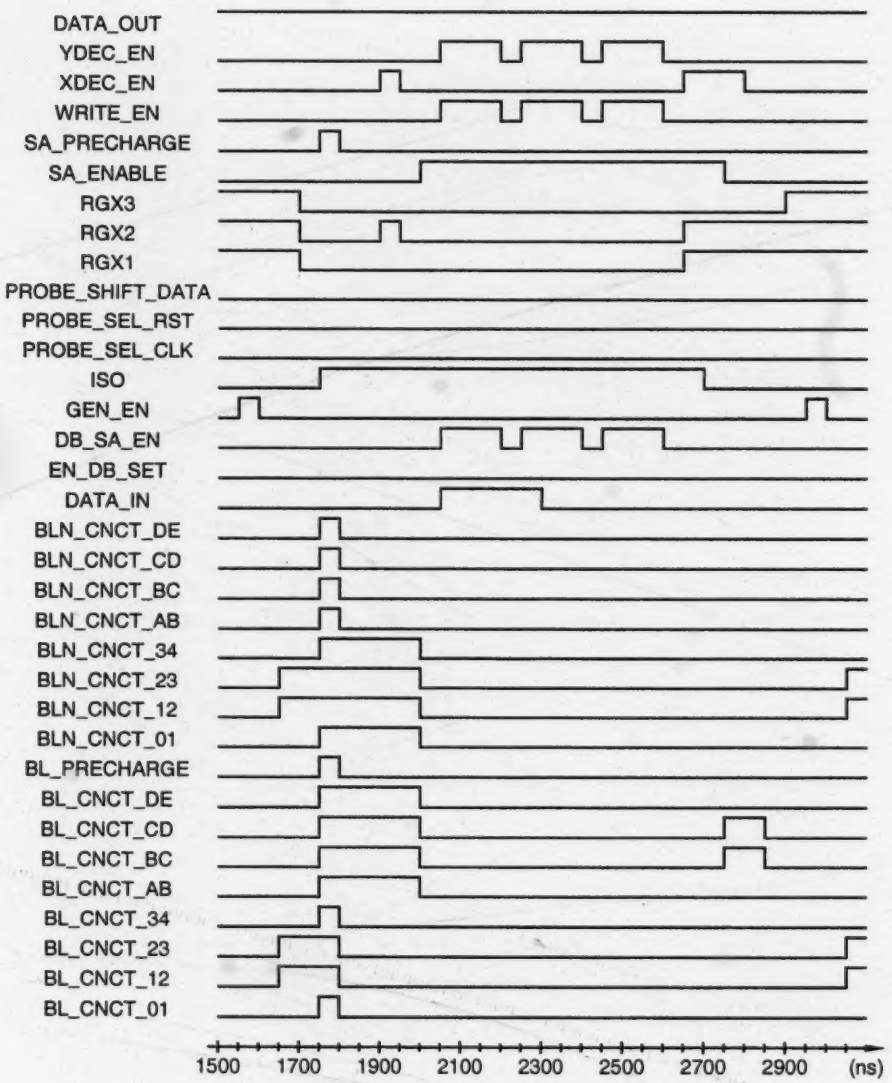


Figure 5.13: ML6 four-level operation write waveforms

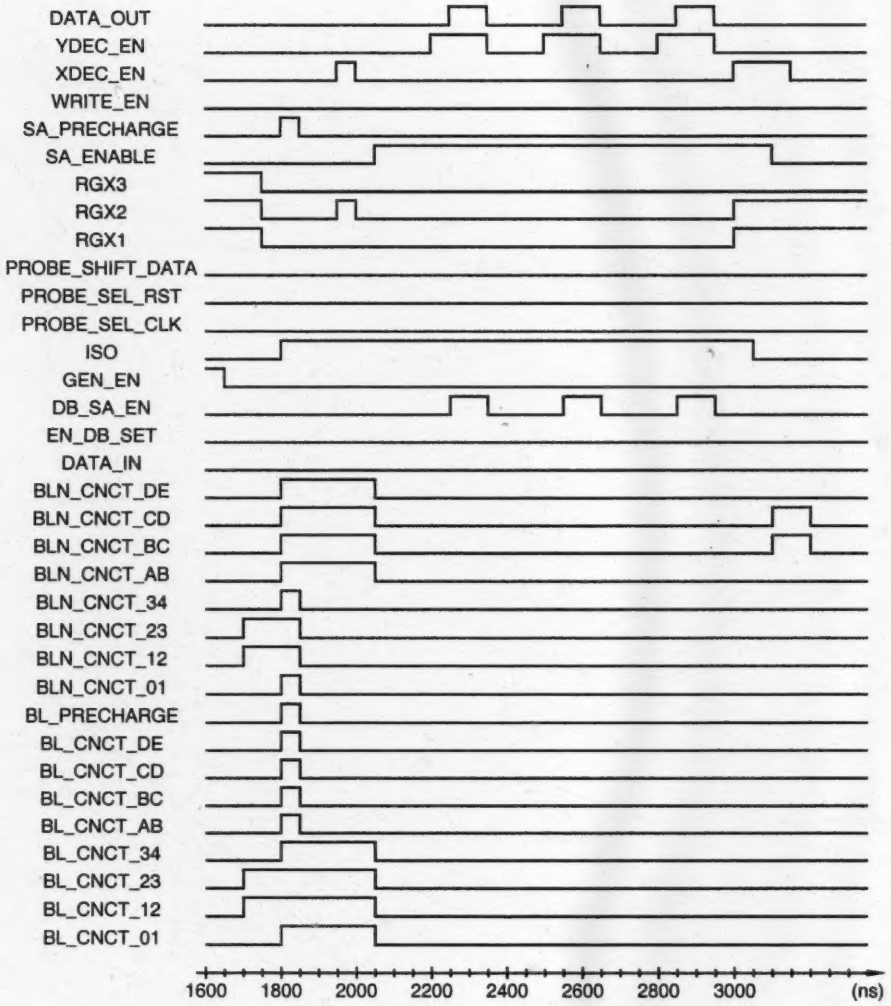


Figure 5.14: ML6 four-level operation read waveforms

5.1.7 Five-level Operation

Five-level operation on ML6 is a straightforward extension of three-level and four-level operation. All signal sequences remain the same as for three-level and four-level operation except for the bitline-connect switches during reference generation, cell dump and restore operations.

For five-level operation, there are two possible reference generation voltage combinations on the switch matrix. The reference voltages that are generated from one of the two 4-by-4 arrays are $\frac{1}{8}V_{DD}$, $\frac{3}{8}V_{DD}$, $\frac{5}{8}V_{DD}$ and $\frac{7}{8}V_{DD}$. The two possible 4-by-4 arrays consist of sections B, C, D and E across bitlines 0, 1, 2 and 3, and sections A, B, C and D across bitlines 1, 2, 3 and 4. Either of the two 4-by-4 arrays will produce the correct reference voltages for five-level operation.

For five-level operation, the four different sections along the addressed bitline have to be accessed for proper reading and writing. The valid codes for five-level operation are: 0000, 0001, 0011, 0111 and 1111. Figures 5.15 and 5.16 shows the input and output waveforms for five-level operation.

5.1.8 Six-level Operation

In six-level operation, the full 5-by-5 switch matrix is used for the reference generation, cell dump and restore operations. All other signal sequences used are the same as for the three-level, four-level and five-level operations. The reference voltages for six-level operation are $\frac{1}{9}V_{DD}$, $\frac{7}{10}V_{DD}$, $\frac{5}{10}V_{DD}$, $\frac{3}{10}V_{DD}$ and $\frac{1}{10}V_{DD}$.

As shown in Figures 5.17 and 5.18, a multiple bitline access has to be made to read from all 5 sections for the 5 bits in any one of the following valid codes: 00000, 00001, 00011, 00111, 01111 and 11111.

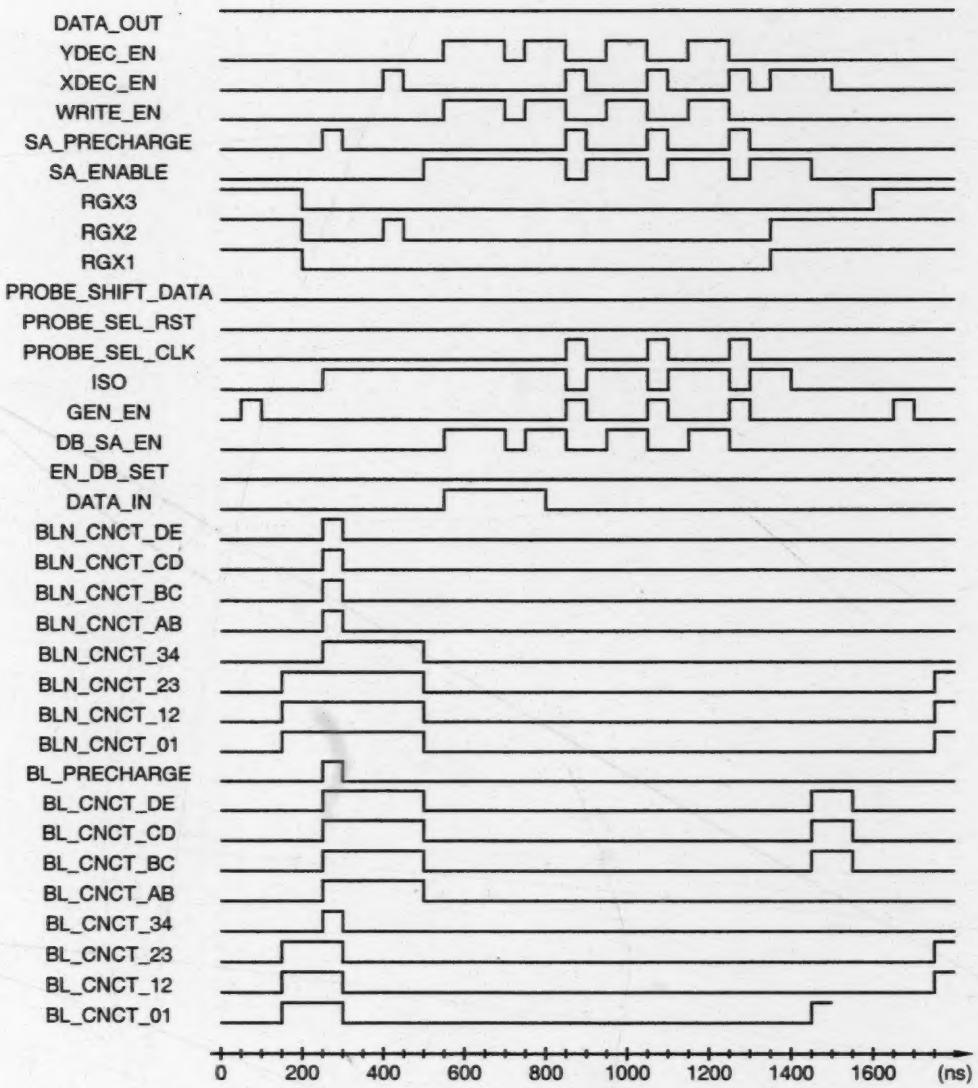


Figure 5.15: ML6 five-level operation write waveforms

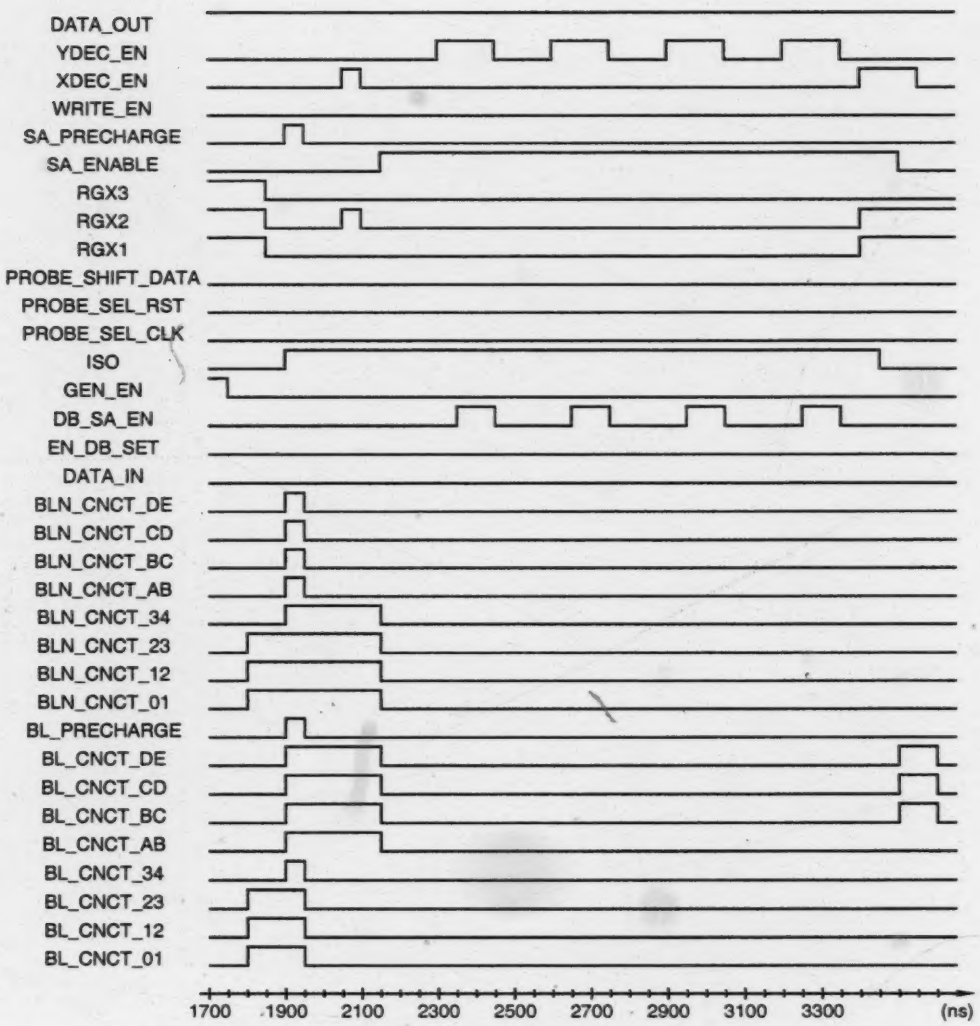


Figure 5.16: ML6 five-level operation read waveforms

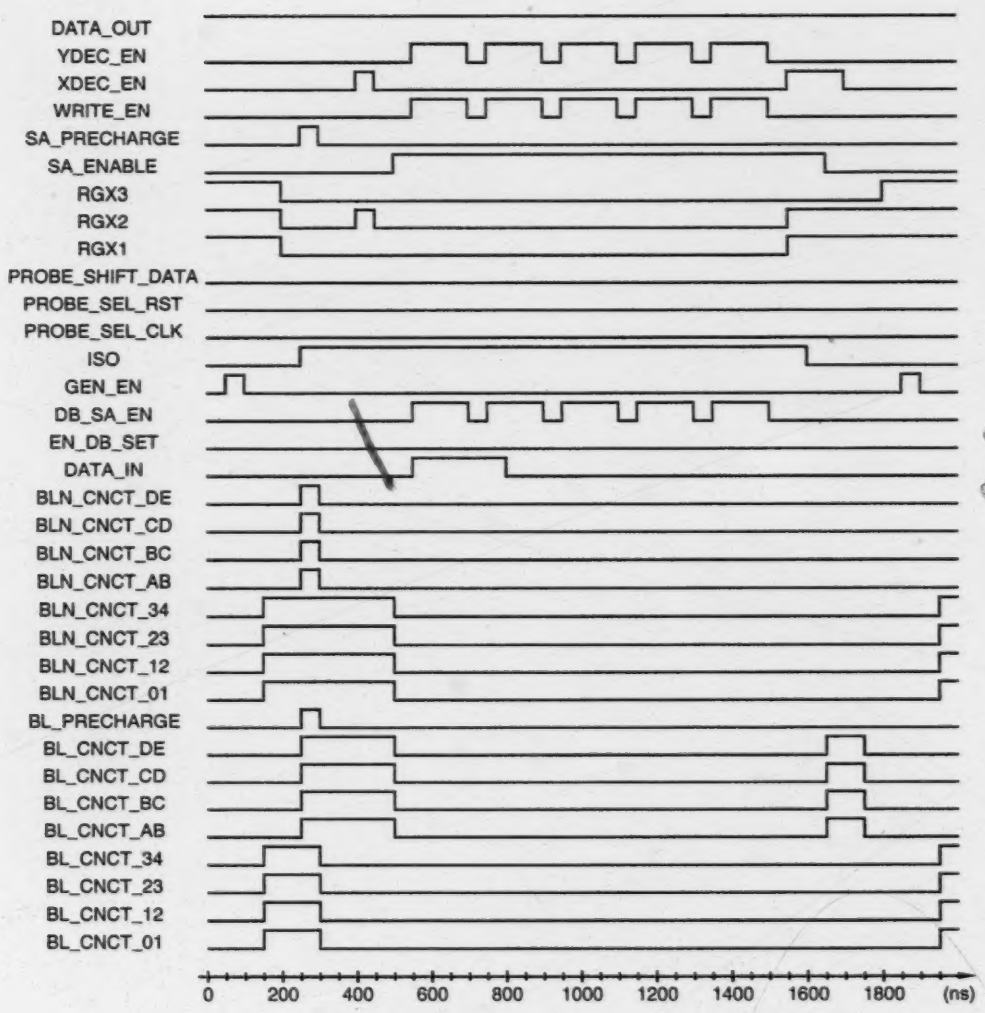


Figure 5.17: ML6 six-level operation write waveforms

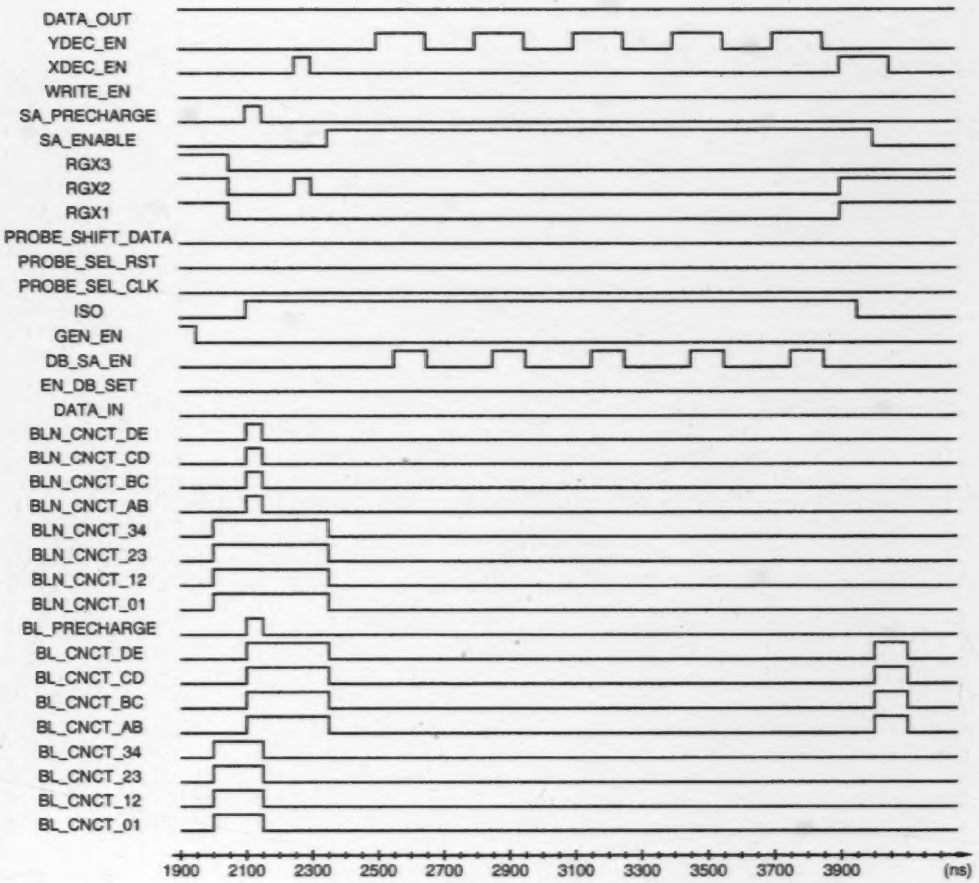


Figure 5.18: ML6 six-level operation read waveforms

5.2 Test Results and Bitmaps

Functional tests of the two-level, three-level, four-level, five-level and six-level operating modes were performed on five test chips. The functional tests were performed on all of the memory cells in all of the chips. Test results showing the percentages of working cells in all levels of operation will be presented in the following sections. Bitmaps showing the distribution of good and non-functional cells will also be presented.

Some examples of tests typically used for memory testing are the zero-one test, checkerboard tests, columns and bars, sliding diagonal, walking one's and zeros, Galloping One's and Zeros Pattern (GALPAT), and the march test [10, 39, 40]. For the purpose of testing the multilevel functionality of ML6, a new test method was devised. Until ML6 is fully characterized, it will not be useful to use conventional memory tests such as the ones mentioned above. A more useful approach would be to tailor the tests according to what is needed to be tested at this point in the research of this MLDRAM design.

Figure 4.5 in Chapter 4 illustrates the test method that is used for ML6. The figure is repeated in Figure 5.19 for convenience. Generally, every memory cell in the chip is tested by writing a data value to the cell and then reading back the value and verifying. For every memory cell, starting from the first bitline (bitline 0) and first wordline (wordline 0) in the first section (section A), running to the last bitline (bitline 319) and last wordline (wordline 12) in the last section (section E), every possible valid code is written and then read back and verified. In six-level operation, for example, the value '00000' is written, read back, then followed by the value '00001' until all the valid codes have been exhausted. In this case, it is '11111'. For the other extreme case of two-level operation, the only valid data is '0' and '1'. So, for two-level operation, a '0' is written and then read back before a'1' is written and read back from a memory cell.

The testing of the memory cells also follow a simple algorithm. For every cell that is to be tested, the code value of '1' is written on the the cell on the next

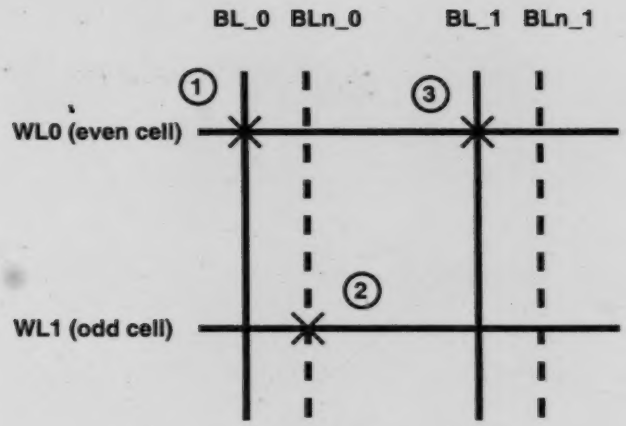


Figure 5.19: ML6 test algorithm

wordline and complement bitline, and also on the cell on the same wordline and the next bitline, before the value on the original cell is read back again. For an example, for six-level mode, the value of '11111' would be written on the other two cells. This algorithm checks for crosstalk that may affect adjacent bitlines, wordlines and cells. This test is important as there is the danger that floating lines (such as databus and bitlines) will float with the correct values and will cause the test to pass and miss catching faults.

5.2.1 Two-level Operation

On both the chips, tests on two-level operation yielded very good results. The percentages of good cells are high. Almost 100% of the memory cells were good. On chip #1, all the cells are two-level operation were good. On chip #2, only one cell (of the tiny cell type) is non-functional. The values in the table shown next were generated using a practical extraction report language (PERL) script written to manipulate output data from the tester. A similar C-program was written for ML5 but the new PERL script is used here instead.

Table 5.1: Chip #1 Yield Matrix for Two-level Operation

Write data level	Read data level	
	0	1
0	100%	0%
1	0%	100%

Table 5.2: Chip #2 Yield Matrix for Two-level Operation

Write data level	Read data level	
	0	1
0	100%	0%
1	0%	100%

5.2.2 Three-level Operation

The functional test results depicted in the table below (for Chip #1) for three-level operation still showed about 99% yield. The reduction in yield compared to two-level operation is about one to three percent.

Table 5.3: Chip #1 Yield Matrix for Three-level Operation

Write data level	Read data level		
	0	1	2
0	100%	0%	0%
1	1%	99%	0%
2	0%	0%	99%

Table 5.4: Chip #2 Yield Matrix for Three-level Operation

Write data level	Read data level		
	0	1	2
0	100%	0%	0%
1	1%	99%	0%
2	0%	1%	99%

5.2.3 Four-level Operation

The yield matrix for four-level operation shows a 6% reduction in yield for some chips for four-level operation. This reduction is expected due to the reduced noise margin that comes with the increase in the number of voltage levels in between V_{SS} and V_{DD} for multilevel sensing. Therefore, a lower yield is expected as the number of allowable voltage levels increases.

Table 5.5: Chip #1 Yield Matrix for Four-level Operation

Write data level	**Read data level**			
	0	1	2	3
0	99%	1%	0%	0%
1	4%	95%	1%	0%
2	1%	2%	96%	1%
3	0%	0%	3%	96%

Table 5.6: Chip #2 Yield Matrix for Four-level Operation

Write data level	**Read data level**			
	0	1	2	3
0	99%	1%	0%	0%
1	4%	95%	1%	0%
2	1%	3%	96%	1%
3	0%	0%	4%	96%

5.2.4 Five-level Operation

As expected, the yield at five-level operation is reduced. The yield is at 86% for some data levels. The yield matrix shows the yield for Chip #1 and #2 for five-level operation.

Table 5.7: Chip #1 Yield Matrix for Five-level Operation

Write data level	Read data level				
	0	1	2	3	4
0	98%	2%	0%	0%	0%
1	11%	87%	2%	0%	0%
2	2%	5%	91%	2%	0%
3	0%	1%	5%	90%	4%
4	0%	0%	1%	10%	89%

Table 5.8: Chip #2 Yield Matrix for Five-level Operation

Write data level	Read data level				
	0	1	2	3	4
0	98%	2%	0%	0%	0%
1	9%	89%	1%	0%	0%
2	2%	5%	91%	2%	0%
3	0%	2%	5%	89%	4%
4	0%	0%	2%	9%	89%

5.2.5 Six-level Operation

The yield matrix shows some degradation in the yield for six-level operation. However, the reduction in yield is not significantly lower than in the case for five-level operation. The worst case yield is 78% for chip #1 and 79% for chip #2.

Table 5.9: Chip #1 Yield Matrix for Six-level Operation

Write data level	**Read data level**					
	0	1	2	3	4	5
0	95%	4%	1%	0%	0%	0%
1	19%	78%	2%	0%	0%	0%
2	5%	9%	83%	3%	0%	0%
3	0%	2%	6%	88%	3%	0%
4	0%	0%	1%	7%	84%	7%
5	0%	0%	0%	3%	14%	83%

Table 5.10: Chip #2 Yield Matrix for Six-level Operation

Write data level	**Read data level**					
	0	1	2	3	4	5
0	97%	3%	0%	0%	0%	0%
1	19%	79%	2%	0%	0%	0%
2	5%	7%	85%	2%	1%	0%
3	0%	2%	7%	86%	4%	2%
4	0%	0%	3%	7%	82%	7%
5	0%	0%	0%	3%	13%	84%

The bitmap in Figure 5.20 is generated for six-level operation.

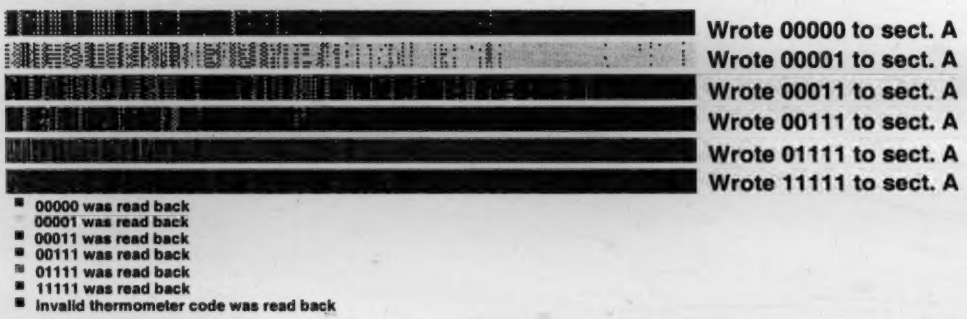

Wrote 00000 to sect. A
Wrote 00001 to sect. A
Wrote 00011 to sect. A
Wrote 00111 to sect. A
Wrote 01111 to sect. A
Wrote 11111 to sect. A
00000 was read back
00001 was read back
00011 was read back
00111 was read back
01111 was read back
11111 was read back
invalid thermometer code was read back

Figure 5.20: ML6 six-level operation bitmap for writing thermometer codes to cells in section A

The bitmap shows geographically, from left to right (bitline 0 to 319), the cell value at a particular time. The first row shows that the data value of '00000' was written onto the cells and read back. Reading back a different value from '00000' would show a different colour shade on the bitmap. Rows two, three, four, five and six show the writing and read back of data values '00001', '00011', '00111', '01111' and '11111' respectively. The bitmap in Figure 5.20 shows the cell values for section A for writing and reading the different thermometer code values for six-level operation. From the top, the first row shows the writing and read back of the value '00000', and so on. The last row would then show the writing and reading of the value '11111' to the cells in section A. Bitmaps for other sections are similar to the one shown in Figure 5.20.

The bitmaps show a clustering of non-functional cells where the tiny cells are. The clustering of the non-functional cells in the area of the tiny cells is expected since the tiny cells would have lesser cell capacitance, thus reduced storage capacity, reduced data retention time and smaller noise margins.

5.2.6 Cell Yield Comparison

Figure 5.21 shows the average cell yield for chips #1 and #2. The worst case yield for the two chips is approximately 86% for six-level operation.

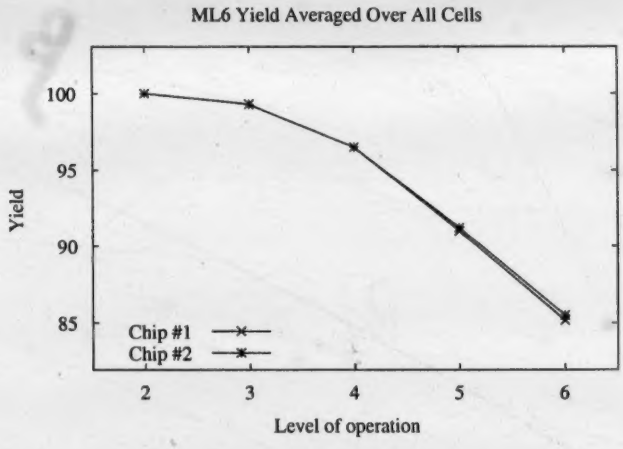
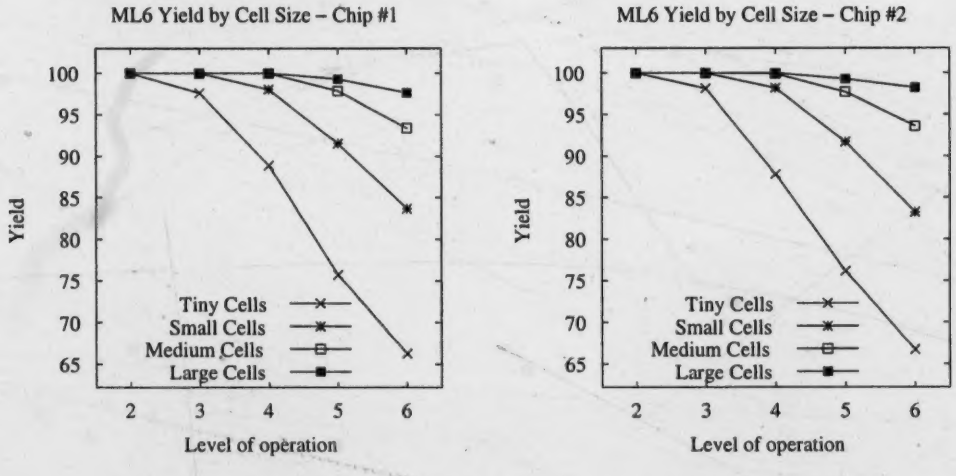


Figure 5.21: ML6 yield averaged over all cells

Figures 5.22(a) and 5.22(b) show the cell yield for the different memory cell sizes for chips #1 and #2. It can be observed that the large cells have the best cell yield, as one might have expected.

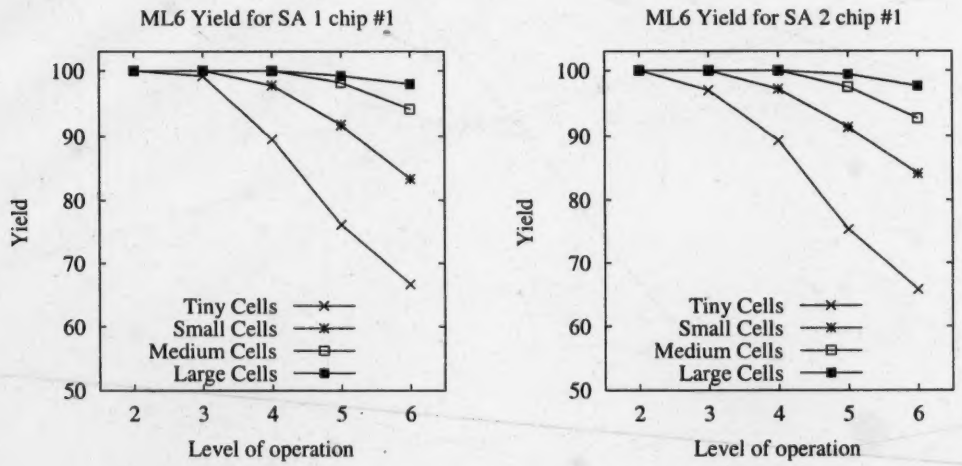


(a) Cell yield for chip #1

(b) Cell yield for chip #2

Figure 5.22: ML6 cell yield for the different memory cell sizes

Figures 5.23(a) and 5.23(b) show the cell yield for the two different sense amplifier sizes in chip #1. Interestingly, the two sense amplifier sizes seem to produce the same cell yields. This suggests that the sense amplifier input offsets which should be smaller for the larger sense amplifier, are not the most important factor affecting cell yield.

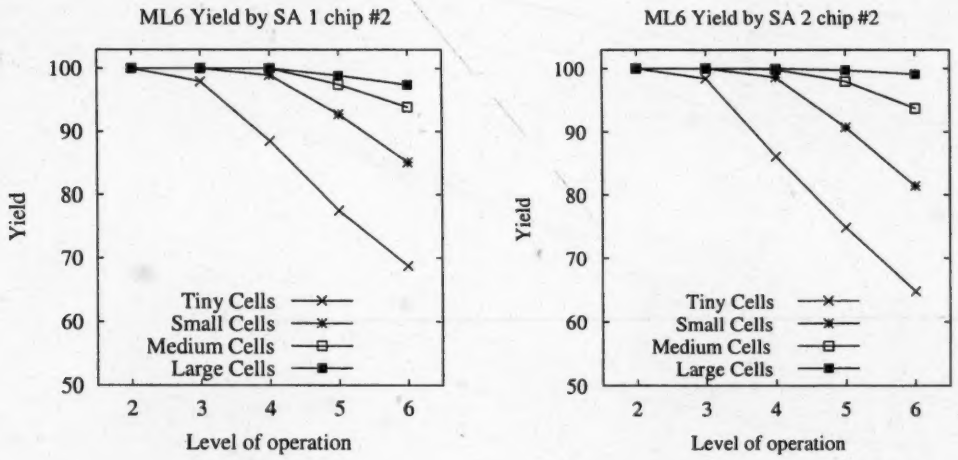


(a) Cell yield for SA1

(b) Cell yield for SA2

Figure 5.23: ML6 cell yield for the different sense amplifier sizes for chip #1

Figures 5.24(a) and 5.24(b) show the cell yield for the two different sense amplifier sizes in chip #2. Unlike in chip #1, the larger sense amplifiers ('SA1') seem to have a slightly higher cell yield.



(a) Cell yield for SA1

(b) Cell yield for SA2

Figure 5.24: ML6 cell yield for the different sense amplifier sizes for chip #2

Figures 5.25(a) and 5.25(b) show the cell yield for the shielded and non-shielded bitlines in chip #1. As expected, the bitline sheilds seem to be effective in reducing bitline-bitline coupling noise, and this increases the cell yield.

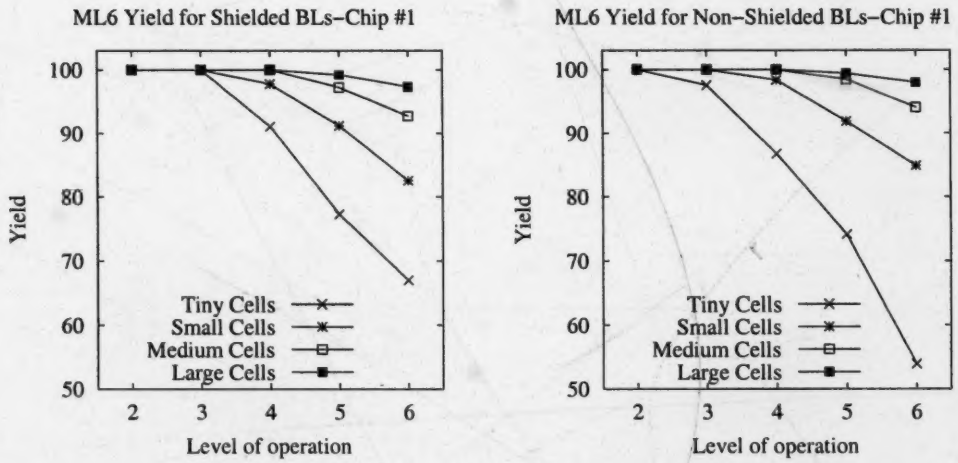
Figures 5.26(a) and 5.26(b) show the cell yield for the shielded and non-shielded bitlines in chip #2.

5.2.7 Temperature Probe

The results in Table 5.11 have been obtained at the output of the core temperature probe at ambient room temperature. An increase in voltage can be seen at the output when the chip is active. For this measurement, the chip was held active for three minutes before the voltage value was read from the oscilloscope.

Table 5.11: Core Temperature Probe at Room Temperature

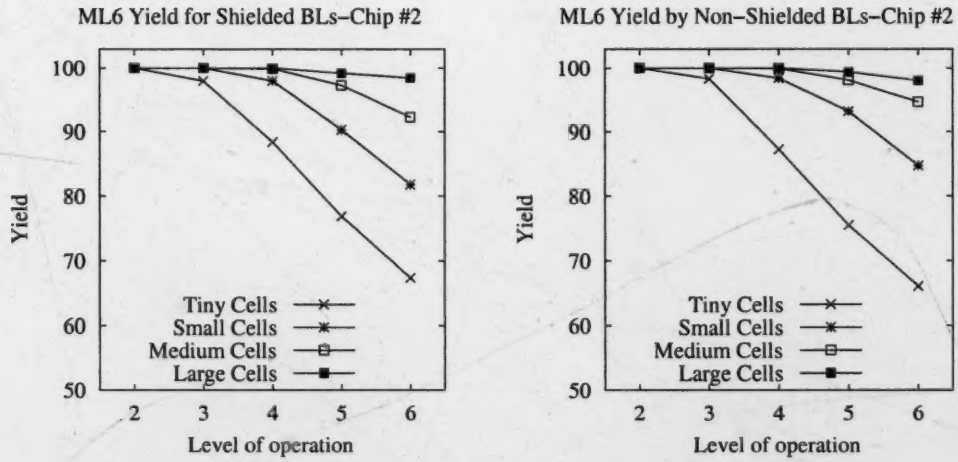
Room temperature	**Temperature probe output**
Chip is idle	750 mV
Chip is active	797 mV



(a) Shielded bitlines

(b) Non-shielded bitlines

Figure 5.25: ML6 cell yield for the shielded and non-shielded bitlines in chip #1



(a) Shielded bitlines

(b) Non-shielded bitlines

Figure 5.26: ML6 cell yield for the shielded and non-shielded bitlines in chip #2

5.3 Discussion

The above yield matrices show evidence that the yield of the memory cells degrades as the number of signal levels increases. This test result is expected since we can understand that the noise margin in between the reference voltage levels decreases as the number of possible data levels increases. The reduced noise margins means less margin for signal errors, say due to noise.

Another phenomenon seen in the test results is the clustering of non-functional cells for the smallest cell size area. Perhaps the charge sharing and balancing in the bitlines during the cell dump and restore operations are not creating the correct values voltage values when the cell charges are small.

The reference and generate wordlines are used to "attach" and "detach" dummy cells to the bitlines for charge balance during the cell dump and restore operations. The reference and generate wordlines are, in turn, controlled by a decoding sequence of logic gates with RGX1, RGX2 and RGX3 as the inputs. It is possible that the RGX1, RGX2 and RGX3 signals are not given enough flexibility to charge balance the sensing circuits properly. There is also a possibility that race conditions could occur when the bitline switches are turned on, causing the sense amplifier to detect the wrong values. The large cells would react to the changes more slowly due to the higher capacitance. There are a number of possible causes of the clustering of non-functional cells in the tiny cells area. One possible cause is the reduced noise margins due to the smaller capacitor size. Another reason could be that the smaller sized capacitors are more leaky due to the shorter transistor channel and gate lengths [32]. Leakage will cause problems in MLDRAMs. Not only will leakage increase power consumption in MLDRAMs, but also reduce data storage retention time and induce more noise into the system. More detailed characterization is required to identify the underlying causes.

The plots showing the cell yield of the shielded and non-shielded bitlines clearly indicate that the cells on the shielded bitlines gave better yield. The plots on the two different sense amplifier sizes, however, do not show a significant difference in the

cell yield for the two sense amplifier sizes. Chip #2 had only a slightly higher cell yield for the larger sense amplifiers. More tests and yield analysis would have to be done to investigate the performance of the two sense amplifier sizes. This will be left for future work.

5.4 Other Relevant Tests

Some other tests were used to test ML5. These tests should also be performed for ML6. One useful test that can be performed on the chip is the cell retention time test. The cell retention time test can be performed by writing data values onto the cells and then reading them back, as in a functional test. Adjusting the delay in between the writing and the reading of the correct data value from the cell will allow the cell retention time to be measured. The cell retention time test can also be modified to measure the leakage current in the cells. The drifting of the cell data can be observed by writing both the lowest and highest data values into the cells, and then observing the drift in data cell voltages in the same cell over time.

Another useful test would be a cell plate bump test. In ML6, the cell plate is directly connected to the back bias, transistor bulk voltage, V_{BB}. Varying the cell plate voltage directly by changing V_{BB} would allow us to observe the effect of changing the cell plate voltage to cell performance. The cell plate, for example, could be bumped in one direction after a cell write to see what effect will be observed at a subsequent cell read. This technique can be used to directly measure the noise margins.

The above mentioned tests and other characterization experiments can be performed with the help of the built-in analog-to-digital (A/D) voltage probes. Using the A/D probes, some useful experiments such as the sense amplifier offset measurements can be performed and studied. Internal circuit nodes can be measured using successive comparisons with known reference voltages, without having physical pins connected to them from outside of the chip.

There are numerous useful experiments that can be performed using the A/D voltage probes. The internal operating temperature on various points on the chip

can also be observed through the built-in temperature probes. In addition to these features, the databus values can be varied and observed for stuck-at faults.

The built-in circuits were added to aid in the characterization of the multilevel DRAM chip in the hope that more insights might be obtained to the precise behaviour of the reference generation and sensing circuits. The results and answers that can be acquired will be invaluable to further multilevel DRAM research.

5.5 Conclusion

It was shown that the functionality of two ML6 die at all levels of operation has been verified. A summary of the problems encountered during chip testing is shown in Table 5.12.

Table 5.12: Summary of Test Effort

Problems Encountered	**Resolutions**
Chip was not working when tested with vectors created for pad-to-pad simulation.	Debugged chip using bottom-up approach – started debugging from databus to sense amplifier and then to memory cell.
Databus sense amplifier and bitline sense amplifier fighting against each other.	I adjusted databus sense amplifier enable signal to be asserted after the bitline sense amplifier is asserted during sensing. Both sense amplifiers can be deasserted at the same time after.
Data cells retain values previous written to them. The tester power down-and-up process did not initialize or delete the old values.	Used bitline precharge to initialize cells before writing to the cells. Bitline precharge enable signal was kept high for a split second after the wordline was asserted for writing into the cell.
Data values read back from the cells are shifted up by one level during multilevel sensing. For more information see Section 5.2.5	I found that I was debugging the tiny cells. So, I used the same vectors on the larger cells. The larger cells worked perfectly for multilevel sensing. Some tiny cells were apparently more leaky or not working.
Erroneous toggling is seen at the databus output.	Tester set up file for the ISO signal was corrupted. A voltage clamp was set for the signal causing the signal to not to toggle at the full value.

Chapter 6

Conclusions

ML6 is a multilevel DRAM chip built for characterization. The new design was motivated by the experience gained from a previous MLDRAM chip, ML5, designed by MSc student Yunan Xiang [42]. ML5 was in turn based on Birk's MLDRAM [5]. The circuits added to aid in the characterization and data collection include the A/D voltage probes, analog temperature sensor and analog databus voltage set circuit. The NMOS transistor switches in the bitlines and switch matrix were replaced with transmission gates for more accurate switched signals. For the same reason, the input and output signals going into and out from the periphery are buffered at strategic points in the chip. The power consumption for every circuit component was also calculated and the routing metal widths in the chip layout were adjusted accordingly to accommodate the required current density. In the memory array, the cell sizes are varied (large, medium, small and tiny) with two sizes of sense amplifiers for each cell size. In the periphery circuit, the address decoder is able to automatically increment, decrement and hold addresses for fast page-mode access to the cells.

6.1 Extracted Layout Simulation versus Actual Chip

ML6 was designed from TSMC's 0.18-μm process technology. Analog simulation was done pad-to-pad to verify the correct operation of a small core extracted from the layout of the chip. The reduced core is ten bitlines wide with five sections

and 12 wordlines in each section. When the fabricated chips were received, the vectors used in the simulation were converted to test vectors and used with the VLSI lab's HP 81200 VLSI tester. Although relaxed timing (50 MHz) was used, the simulation vectors did not immediately work on the real chip. Many adjustments had to be made to the edges of the switching signals. The databus and bitline sense amplifiers seemed to be fighting one another at one point, preventing the databus from containing the current values from the addressed bitline.

Simulation on the extracted layout of the chip was not able to show the need to initialize or precharge the memory cells before writing at the beginning of a test. When the first chip was tested, the test had to be run more than two times before the values on the addressed chip stabilized. It was inferred from this behavior that the cells needed to be first precharged to $\frac{1}{2}V_{DD}$ during the initial bitline and sense amplifier precharge to clear the cells of any previous data values before the first write.

6.2 Chip Evaluation

After debugging the test program, chips 1 and 2 were found to have a relatively high cell yield for all levels of operation. Six-level operation for the first two chips yielded almost 86% good cells.

One pattern of cell failures that could be seen immediately from the initial functional tests is that the non-functional[1] cells are concentrated in the tiny cell size area. The functional test can be modified so that all cells are initialized during the bitline and sense amplifier precharge before any test sequence is started. In this way, the old values in the cells are erased (precharged to $\frac{1}{2}V_{DD}$) before the start of the test. Then, perhaps more cells from the large cell size area would pass the functional test. Precharging the cells to $\frac{1}{2}V_{DD}$ will help to reduce the sensing time for the large cells since they need to only charge or discharge from half of the voltage magnitude to reach the V_{DD} or V_{SS} data voltage.

[1]Non-functional cells are defined as the cells of which the data written are read back with error.

6.3 Suggestions for Improvement

Some of the problems faced during debug could have been avoided had the design been done differently. One problem that was immediately seen during the initial debug was the databus and bitline sense amplifiers driving against one another. Although this problem was solved by changing the timing sequence between the critical sensing signals, the problem could have been avoided if the databus sense amplifier had been made larger than the bitline sense amplifiers. There are no isolation switches to disconnect the databus from the bitlines when YDEC_EN is on. So for the split second when both the databus and bitline sense amplifiers are on, the imbalance in the databus and the addressed bitline could cause the sense amplifiers to amplify the data value to the wrong voltage rail. Also, the write circuitry on the databus could be made stronger with larger transistor sizes so that the input data values can overdrive the databus and bitline voltages to write to a cell. Improvements in both the databus sense amplifiers and the data-input write circuitry would greatly improve the writing of data into the cells. Reading from the cells would also be improved since the stronger databus sense amplifiers would be able to detect the small change in the voltage on the addressed databus and assist the bitline sense amplifier in amplifying the data voltage to the correct value. Presently, the drive power of the databus sense amplifier and the write circuit is adequate but there is room for improvement in this area.

Another improvement that could be done is the disconnection of the databus precharge signal from the YDEC_EN signal. The YDEC_EN signal is accessible from outside of the chip but the DB_PRECHARGE signal is a delayed and inverted version of the YDEC_EN signal. The degree of control flexibility is reduced when the databus precharge signal is controllable only from the YDEC_EN signal. For the characterization of the chip, it may be necessary to be able to turn off the Y address decoding while the databus precharge is also turned off. This feature would come in handy if we were to want to precharge the databuses to $\frac{1}{2}V_{DD}$ using the analog databus-set circuit or to put any analog value onto the databus using the databus-set

circuit. Presently, however, the internal circuit delay built for the databus precharge signal has adequate delay for the chip to function properly for all operation modes.

It is suspected that the RGX1, RGX2 and RGX3, reference and generate wordline control signals are not balancing the bitlines properly during sensing. Breaking up the RGX signals into their respectively reference and generate control signals would require too many pins as that would be adding at least four extra pins for every section that is present. Perhaps the reference wordline control signals could be decoded to save pins and the generate wordlines could be left as pins accessible from outside of the chip. Again, the present chip functions in all levels of operation with the RGX1, RGX2 and RGX3 signals.

With the A/D voltage probes in place, a full characterization of the existing chip can be carried out even without the changes mentioned in the preceding paragraphs.

6.4 Accomplishments in this Thesis Research

The design, evaluation and functional testing of ML6 took almost thirteen months to complete. The design and layout stage spanned over a period of seven months while the fabrication of the chips took about six months to complete. It has taken me approximately three months to debug the test chip and to create working test vectors for all levels of operation for the multilevel DRAM. Other students have helped and participated in the design process while I was responsible for the schematic design and simulation of all the circuits in the chip, the design and layout of the temperature probe, and the functional testing and debug of the test chip. I believe that my ability to have debugged and tested the chip I have designed is an achievement in itself. This is the first time in the history of MLDRAM research at the University of Alberta that a student is also able to successful debug and test a chip designed during the course of his/her MSc research program.

I have included a summary of my accomplishments made during the design and simulation of ML6:

Analog databus voltage set circuit I have designed the circuit based on a sugges-

tion by my supervisor, created the schematic, sized the transistors and simulated the circuit to ensure proper operation using Cadence IC design tools suite. The circuit have proven to have worked on the real chip when I used it to precharge the databus to $\frac{1}{2}V_{DD}$ for an experiment while debugging the databus functionality on the chip.

A/D voltage probes Based on my supervisor's idea, I have designed and simulated this circuit. Several iterations of the design have been made and the successes of the design presented to my supervisor. Finally, the transistors in the complete design are properly sized for better results.

Bitline sense amplifier The sense amplifiers in ML5 were precharged together with the bitlines due to misplaced isolation transistors in between the sense amplifiers and the bitline segments. I have analyzed the sense amplifier circuit in ML5 and modified the circuitry to be more simple and robust in that they sense amplifiers can be isolated, precharged and used independently from the bitline segments. The sense amplifier activation is made more simple by directly connecting the sense amplifier enable signals to an externally controllable input. Based on a suggestion by my supervisor, I have simulated and sized two sets of sense amplifier design sizes for the purpose characterizing the behavior and performance of various sense amplifier sizes with respect to various cell sizes.

Databus sense amplifier A sense amplifier was added to all databus pairs to improve sensing. The databus circuitry was based on a idea from a colleague. The databus sense amplifier and write driver circuits were simulated and sized appropriately. The databus worked perfectly in the test chip.

Address decode improvement over ML5 I have added descrambling and modified the address decoders in ML5 for pin reduction and automatic address generation for page mode operations. The designs of the periphery were

made from schematic entry and Verilog implementation of the decoding logic.

Improved ML5's column-decode-databus-precharge A race condition occurred in ML5's column decode and databus precharge circuit due to insufficient delay in the column decode enable signal. I proposed using an external pin to control the databus precharge but the idea was discarded due to the need to reduce pin count. I fixed the race condition problem in ML6 by adding the appropriate delay in the circuit through careful delay calculations and simulations.

Memory core and pad-to-pad simulation The full schematic of the test chip was created and a reduced core was simulated in the analog environment for all levels of operation in ML6. Additionally, the pad-to-pad simulation was also performed with the reduced core and periphery components added.

Temperature probe A CMOS temperature probe was sized, simulated and laid out for ML6. The design was decided upon after literature search in the area of analog temperature probes.

Switch matrix using transmission gates Based on a suggestion by my supervisor, transmission gates were used in the switch matrix and in place of transfer switches. The comparison in performance between the transmission and NMOS transfer switch in terms of charge-injection cancellation and source-drain voltage drop are made via circuit simulation. Due to my discovery, charge-injection cancellation transistors are not needed in the design. The use of transmission gates is enough to cancel out charge-injection due to switching.

Bitline-cell capacitance/power calculations Capacitances between the cells and bitlines were calculated and circuit simulated to determine the appropriate cell sizes for ML6. The same calculations were done in ML5 for com-

parison. The power consumption of each circuit in the ML6 test chip was also calculated to determine the appropriate wire size for circuit layout.

The following list summarizes my accomplishments made during chip debugging and testing:

Databus functionality debug I debugged and made sure all databus lines can be driven fully to V_{SS} and V_{DD}. Every single databus was addressed and sucessfully driven to the appropriate values using the write driver.

Sense amplifier functionality debug I checked the functionality of the sense amplifiers by writing and then reading from them without accessing the cells. This proves the bitline sense amplifier's functionality

Cell functionality debug One cell from every cell size group was written to and read from to check for functionality. Timing between control signals are adjusted to ensure robust and stable operation.

Debugged test chip for two, three, four, five and six-level operation I debugged the test chip and adjusted signal timings to function for all possible levels of operation for ML6. All functional vectors are now in working order and ready for testing for the rest of the packaged chips.

Temperature probe circuit I have successfully recorded the voltage corresponding to the ambient temperature of the memory core and periphery of the chip during the active (while some tests are running) and inactive (idle) modes.

6.5 Future Work

Much can be learned from the characterization of ML6. It is when we first understand the real practical challenges in multilevel sensing well enough that improvements can be made to the chip and the circuits within. The full and extensive characterization of ML6 would answer many questions with leakage currents, voltage

offsets and the operational speed and conditions that we have now. New multilevel DRAM architectures and fault models can be developed from these answers.

The wealth of knowledge that could be obtained from this chip would be invaluable to further MLDRAM research. It is my fervent hope and desire that one day multilevel DRAMs will find a place in the niche market of embedded DRAMs and file store memories.

Bibliography

[1] M. Aoki, Y. Nakagome, M. Horiguchi, S. Ikenaga, and K. Shimohigashi. A 16-level/cell dynamic memory. *IEEE Journal of Solid-State Circuits*, SC-22(2):297–299, April 1987.

[2] M. Aoki, Y. Nakagome, M. Horiguchi, H. Tanaka, S. Sunami, and K. Itoh. A 60-ns 16-Mbit CMOS DRAM with a transposed data-line structure. *IEEE Journal of Solid-State Circuits*, 23:1113–1119, October 1989.

[3] R. J. Baker, H. W. Li, and D. E. Boyce. *CMOS: Circuit Design, Layout, and Simulation*. IEEE Press, 1998.

[4] A. Bakker and Johan H. Huijsing. Micropower CMOS temperature sensor with digital output. *IEEE Journal of Solid-State Circuits*, 31(7):933–937, July 1996.

[5] G. Birk. Evaluation, design and implementation of multilevel DRAM. Master's thesis, University of Alberta, 1999.

[6] G. Birk, B. F. Cockburn, and D. G. Elliott. A comparative simulation study of four multilevel DRAMs. *IEEE Internal Workshop on Memory Technology, Design and Testing*, pages 102–109, August 1999.

[7] M. T. Bohr. Nanotechnology goals and challenges for electronic applications. *IEEE Trans. on Nanotechnology*, 1(1):56–62, Mar 2002.

[8] A. Chan. Design and implementation of a multilevel DRAM. Master's thesis, University of Alberta, 2000.

[9] CMC. PGA68 pin bonding and layout diagrams. `http://www.cmc.ca/prod_serv/des_fab_test/packaging/68pga_bond.html`, 2001.

[10] B. F. Cockburn. Tutorial on semiconductor memory testing. *Journal of Electronic Testing: Theory and Applications*, pages 321–336, May 1994.

[11] Intel Corporation. Intel executive bio. `http://www.intel.com/pressroom/kits/bios/moore.htm`, 2003.

[12] T. Kawahara et al. A high-speed, small-area, threshold-voltage-mismatch compensation sense amplifier for gigabit-scale DRAM arrays. *IEEE Journal of Solid-State Circuits*, 28(7):816–823, July 1993.

[13] T. Furuyama, T. Ohsawa, Y. Nagahama, H. Tanaka, Y. Watanabe, T. Kimura, and K. Muraoka. An experimental 2-bit/cell storage DRAM for macrocell or memory-on-logic application. *IEEE Journal of Solid-State Circuits*, 24(2):388–393, April 1989.

[14] P. Gillingham. A sense and restore technique for multilevel DRAM. *IEEE Transactions on Circuits and Systems-II: Analog and Digital Signal Processing*, 43(7):483–486, July 1996.

[15] T. P. Haraszti. *CMOS Memory Circuits*. Kluwer Academic Publishers, 2000.

[16] H. Hidaka, Y. Matsuda, and K. Fujishima. A divided/shared bit-line sensing scheme for ULSI DRAM cores. *IEEE Journal of Solid-State Circuits*, 26:473–477, April 1991.

[17] M. Inoue, H. Kotani, T. Yamada, H. Yamauchi, A. Fujiwara, J. Matsushima, H. Akamatsu, M. Fukumoto, M. Kubota, I. Nakao, N. Aoi, G. Fuse, S. Ogawa, S. Odanaka, A. Ueno, and H. Yamamoto. A 16Mb DRAM with an open bit-line architecture. *IEEE ISSCC Digest of Technical Papers*, pages 246–247, 1988.

[18] K. Itoh. *VLSI Memory Chip Design*. Springer, 2001.

[19] S. Kang and Y. Leblebici. *CMOS Digital Integrated Circuits: Analysis and Design*. McGraw Hill, 2003.

[20] B. Keeth and R. J. Baker. *DRAM Circuit Design: A Tutorial*. IEEE Press, 2001.

[21] R. Kraus and K. Hoffmann. Optimized sensing scheme of DRAMs. *IEEE Journal of Solid-State Circuits*, 24:895–899, August 1989.

[22] Y. Nakagome, M. Aoki, S. Ikenaga, M. Horiguchi, S. Kimura, Y. Kawamoto, and K. Itoh. The impact of data-line interference noise on DRAM scaling. *IEEE Journal of Solid-State Circuits*, 23:1120–1127, October 1988.

[23] T. Okuda and T. Murotani. A four-level storage 4-Gb DRAM. *IEEE Journal of Solid-State Circuits*, 32(11):1743–1747, November 1997.

[24] Compaq Computer Organization and ISSG Technology Communications. Memory technology evolution: An overview of system memory technologies. Technical Report TC010603TB, June 2001.

[25] B. Prince. *Semiconductor Memories: A Handbook of Design, Manufacture, and Application*. John Wiley, 2nd edition, 1996.

[26] J. M. Rabaey, A. Chandrakasan, and B. Nikolic. *Digital Integrated Circuits: A Design Perspective*. Prentice Hall, 2003.

[27] A. K. Sharma. *Advanced Semiconductor Memories: Architectures, Designs, and Applications*. John Wiley, 2003.

[28] V. Szekely, Cs. Marta, Zs. Kohari, and M. Rencz. New temperature sensors for DfTT applications. *THERMINIC Workshop*, pages 49–55, September 1996.

[29] V. Szekely, M. Rencz, and B. Courtois. Integrating on-chip temperature sensors into DfT schemes and BIST architectures. *Proceedings of the 15th IEEE VLSI Test Symposium*, pages 440–446, April 1997.

[30] D. Takashima, S. Watanabe, H. Nakano, Y. Oowaki, and K. Ohuchi. Open/-folded bit-line arrangement for ultra-high-density DRAM's. *IEEE Journal of Solid-State Circuits*, 29:539–542, April 1994.

[31] Thermosens. Thermosens: Temperature Sensor. `http://www.eet.bme.hu/prosoma/thermosens/descriptionx.html`, 2000.

[32] S. Thompson, P. Packan, and M. Bohr. MOS scaling: Transistor challenges for the 21st century. *Intel Technology Journal Q3'98*, pages 1–19, Quarter 3 1998.

[33] TSMC. *Virtual Silicon Technology Inc. Native-18 Standard Cell Library 0.18 μm TSMC Process*, 1999.

[34] TSMC. *TSMC 0.18-μm logic 1P6M Salicide 1.8V/3.3V Design Rule*, 2000.

[35] TSMC. *TSMC 0.18 μm Mixed Signal 1P6M Salicide 1.8V/3.3V Spice Models*, 2000.

[36] TSMC. *TSMC 0.18 μm Mixed signal/RF 1P6M+ Salicide 1.8V/3.3V Design Rule*, 2000.

[37] TSMC. *TSMC Al bond pad design rule*, 2001.

[38] M. Tuthill. A switched current, switched capacitor temperature sensor in 0.6-μm CMOS. *IEEE Journal of Solid-State Circuits*, 33(7):1117–1122, July 1998.

[39] A. J. van de Goor. *Testing Semiconductor Memories: Theory and Practice.* John Wiley & Sons, 1991.

[40] P. K. Veenstra, F. P. M. Beenker, and J. J. M. Koomen. Testing of random access memories: Theory and practice. *IEE Proceedings Part G — Electronics Circuits and Systems*, 135:24–28, February 1988.

[41] H. P. Wong, P. M. Solomon, and J. J. Welser. Nanoscale CMOS. In *Proc. of the IEEE*, pages 537–558, April 1999.

[42] Y. Xiang. Design, implementation and testing of a multilevel DRAM with adjustable cell capacity. Master's thesis, University of Alberta, 2002.

[43] Y. Xiang, B. F. Cockburn, and D. G. Elliott. Design of a multilevel DRAM with adjustable cell capacity. *IEEE Canadian Journal of Electrical and Computer Engineering*, 26(2):55–59, April 2001.

[44] T. Yoshihara, H. Hidaka, Y. Matsuda, and K. Fujishima. A twisted bitline technique for Multi-Mb DRAMs. *IEEE ISSCC Digest of Technical Papers*, pages 238–239, October 1988.

Appendix A

ML6 Pinlist, Tester Connections and Pin Description

Table A.1: ML6 Pinlist and Tester Connections

Pin #	Pin Name	Pogo	Color	DUT	Type	Scope
1	WRITE_EN	R1-01	white	C1M 7C1	I–Digital	5C11
2	BL_CNCT_01	R1-03	white	C1M 7C2	I–Digital	2C0
3	BL_CNCT_12	R1-05	white	C1M 7C3	I–Digital	2C1
4	BL_CNCT_23	R1-07	white	C1M 7C4	I–Digital	2C2
5	BL_CNCT_34	R2-01	white	C1M 7C5	I–Digital	2C3
6	ADDR[0]	R2-03	yellow	C1M 7C6	I–Digital	5C0
7	ADDR[1]	R2-05	yellow	C1M 7C7	I–Digital	5C1
8	VDD	DPS1	red		Power	
9	PROBE_RST				I–Digital	
10	TEMP_PERI	R10-01	blue		O–Analog	1C0
11	VSS	GND	black		Ground	
12	ADDR[2]	R1-12	yellow	C1M 3C3	I–Digital	5C2
13	ADDR[3]	R1-14	yellow	C1M 3C7	I–Digital	5C3
14	BLn_CNCT_01	R1-16	white	C1M 3C8	I–Digital	2C4
15	BLn_CNCT_12	R3-12	white	C1M 5C4	I–Digital	2C5
16	BLn_CNCT_23	R3-10	white	C1M 4C1	I–Digital	2C6
17	BLn_CNCT_34	R2-10	white	C1M 8C1	I–Digital	2C7
18	NC					
19	SA_ENABLE	R3-16	white	C1M 5C6	I–Digital	5C12
20	SA_PRECHARGE	R3-14	white	C1M 4C2	I–Digital	5C13
21	BL_PRECHARGE	R2-12	white	C1M 8C2	I–Digital	5C14
22	VDD_RING	DPS2	red		Power	
23	VPP	DPS5	red	R10-16	Power	
24	VBB	DPS4	red		Power	

continued on next page

Table A.1: *continued*

Pin #	Pin Name	Pogo	Color	DUT	Type	Scope
25	PROBE_OUT	R6-14	blue	C1M 3C2	O–Digital	1C1
26	VBLP	DPS3	red		Power	
27	PROBE_VREF	DPS6	blue	R10-10	I–Analog	
28	PROBE_EXTV	DPS7	blue	R10-12	I–Analog	
29	V_ANALOG	DPS8	blue	R10-14	I–Analog	
30	VSS_RING	GND	black		Ground	
31	EN_DB_SET				I–Digital	
32	ISO	R4-12	white	C1M 4C4	I–Digital	1C5
33	NC					
34	YDEC_EN	R4-14	white	C1M 5C1	I–Digital	1C6
35	XDEC_EN	R2-14	white	C1M 8C3	I–Digital	1C7
36	BL_CNCT_AB	R3-05	white	C1M 8C5	I–Digital	2C8
37	BL_CNCT_BC	R2-16	white	C1M 8C4	I–Digital	2C9
38	BL_CNCT_CD	R5-10	white	C1M 5C2	I–Digital	2C10
39	BL_CNCT_DE	R5-12	white	C1M 7C8	I–Digital	2C11
40	ADDR_DEC[0]	R5-16	yellow	C1M 5C3	I–Digital	5C4
41	ADDR_DEC[1]	R6-01	yellow	C1M 5C5	I–Digital	5C5
42	VSS	GND	black		Ground	
43	TEMP_CORE	R10-03	blue		O–Analog	1C2
44	PROBE_SHIFT				I–Digital	
45	VDD	DPS1	red		Power	
46	ADDR_DEC[2]	R6-05	yellow	C1M 6C2	I–Digital	5C6
47	ADDR_DEC[3]	R6-07	yellow	C1M 6C3	I–Digital	5C7
48	BLn_CNCT_AB	R5-01	white	C1M 6C4	I–Digital	2C12
49	BLn_CNCT_BC	R5-03	white	C1M 6C8	I–Digital	2C13
50	BLn_CNCT_CD	R5-05	white	C1M 6C5	I–Digital	2C14
51	BLn_CNCT_DE	R5-07	white	C1M 6C6	I–Digital	2C15
52	NC					
53	RGX3	R4-01	white	C1M 6C7	I–Digital	1C8
54	RGX2	R4-16	white	C1M 5C7	I–Digital	1C9
55	RGX1	R6-03	white	C1M 6C1	I–Digital	1C10
56	PROBE_CLK	R3-01	white	C1M 8C7	I–Digital	1C3
57	DATA_OUT	R4-05	green	C1M 3C1	O–Digital	5C15
58	VSS_RING	GND	black		Ground	
59	VDD	DPS1	red		Power	
60	VBLP	DPS3	red		Power	
61	VBB	DPS4	red		Power	
62	VPP	DPS5	red		Power	
63	VDD_RING	DPS2	red		Power	
64	CLK	R1-10	white	C1M 3C5	I–Digital	5C8

continued on next page

Table A.1: *continued*

Pin #	Pin Name	Pogo	Color	DUT	Type	Scope
65	GEN_EN	R4-10	white	C1M 4C3	I–Digital	1C4
66	DATA_IN	R3-03	green	C1M 8C8	I–Digital	5C9
67	NC					
68	DB_SA_EN	R3-07	white	C1M 8C6	I–Digital	5C10

WRITE_EN Enables data write circuit on databus

BL_CNCT_01 Connects bitlines 0 and 1 in the switch matrix

BL_CNCT_12 Connects bitlines 1 and 2 in the switch matrix

BL_CNCT_23 Connects bitlines 2 and 3 in the switch matrix

BL_CNCT_34 Connects bitlines 3 and 4 in the switch matrix

ADDR[0] Address, bit 0

ADDR[1] Address, bit 1

ADDR[2] Address, bit 2

ADDR[3] Address, bit 3

ADDR_DEC_SEL[0] Selects the type of operation for the four address bits, bit 0

ADDR_DEC_SEL[1] Selects the type of operation for the four address bits, bit 1

ADDR_DEC_SEL[2] Selects the type of operation for the four address bits, bit 2

ADDR_DEC_SEL[3] Selects the type of operation for the four address bits, bit 3

PROBE_SEL_RST Resets probe select counter

TEMP_PERIPHERY_PROBE Temperature probe voltage output from chip periphery

BLn_CNCT_01 Connects complementary bitlines 0 and 1 in the switch matrix

BLn_CNCT_12 Connects complementary bitlines 1 and 2 in the switch matrix

BLn_CNCT_23 Connects complementary bitlines 2 and 3 in the switch matrix

BLn_CNCT_34 Connects complementary bitlines 3 and 4 in the switch matrix

SA_ENABLE Enables bitline sense amplifiers

SA_PRECHARGE Enables sense amplifier precharge circuits

BL_PRECHARGE Enables bitline precharge circuits

PROBE_DIGITAL_OUT Digital output of A/D probe

PROBE_VREF_IN Input reference voltage for A/D probe

PROBE_EXTV External input voltage to check probe functionality

V_ANALOG External analog voltage used to set databuses to the voltage value

EN_DB_SET Enables databus analog voltage set circuit

ISO Isolates sense amplifiers from bitline pairs

YDEC_EN Enables column decoding — also activates databus precharge when off

XDEC_EN Enables wordline decoding

BL_CNCT_AB Connects bitlines in between sections A and B in the switch matrix

BL_CNCT_BC Connects bitlines in between sections B and C in the switch matrix

BL_CNCT_CD Connects bitlines in between sections C and D in the switch matrix

BL_CNCT_DE Connects btilines in between sections D and E in the switch matrix

TEMP_CORE_PROBE Temperature probe voltage output from chip core

PROBE_SHIFT_CLK Signal to shift select register for A/D probe

BLn_CNCT_AB Connects complementary bitlines in between sections A and B in the switch matrix

BLn_CNCT_BC Connects complementary bitlines in between sections B and C in the switch matrix

BLn_CNCT_CD Connects complementary bitlines in between sections C and D in the switch matrix

BLn_CNCT_DE Connects complementary bitlines in between sections D and E in the switch matrix

RGX3 Multiplexed reference and generate wordline activate signal – asserts all reference and generate wordlines

RGX2 Multiplexed reference and generate wordline activate signal – asserts reference wordlines (RWLs) in all sections for dumping reference cell voltages onto the full length complement bitlines

RGX3 Multiplexed reference and generate wordline activate signal – asserts even or odd reference and generate wordlines in sections other than the section containing the addressed wordline

PROBE_SEL_CLK Clock to probe select register

DATA_OUT Data output of chip

CLK Clock to latch in the four address bits

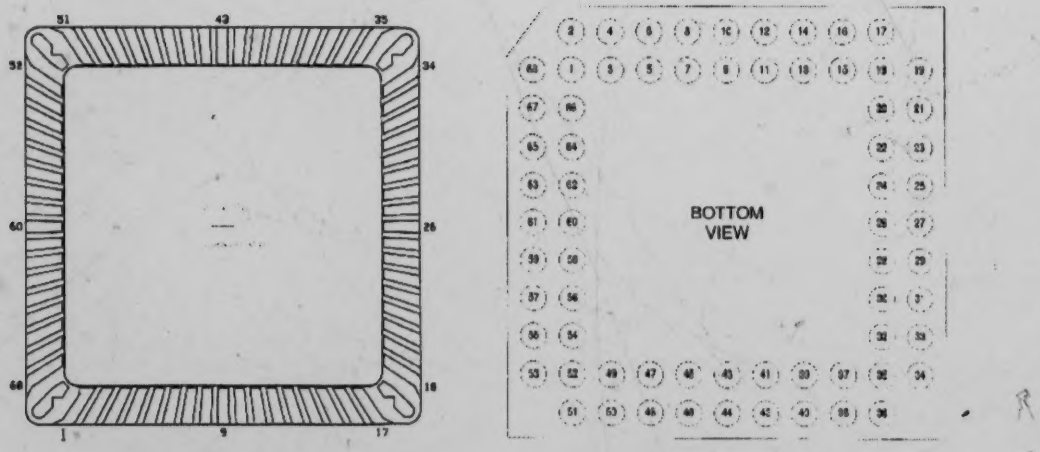
GEN_EN Enables reference generation for multilevel sensing

DATA_IN Data input of chip

DB_SA_EN Enables databus sense amplifiers for sensing

Appendix B

ML6 PGA68 Pin Bonding and Layout Diagrams



(a) PGA68 bonding diagram

(b) PGA68 pin out diagram (bottom view)

Figure B.1: PGA68 layout and pin bonding

Appendix C

ML6 die

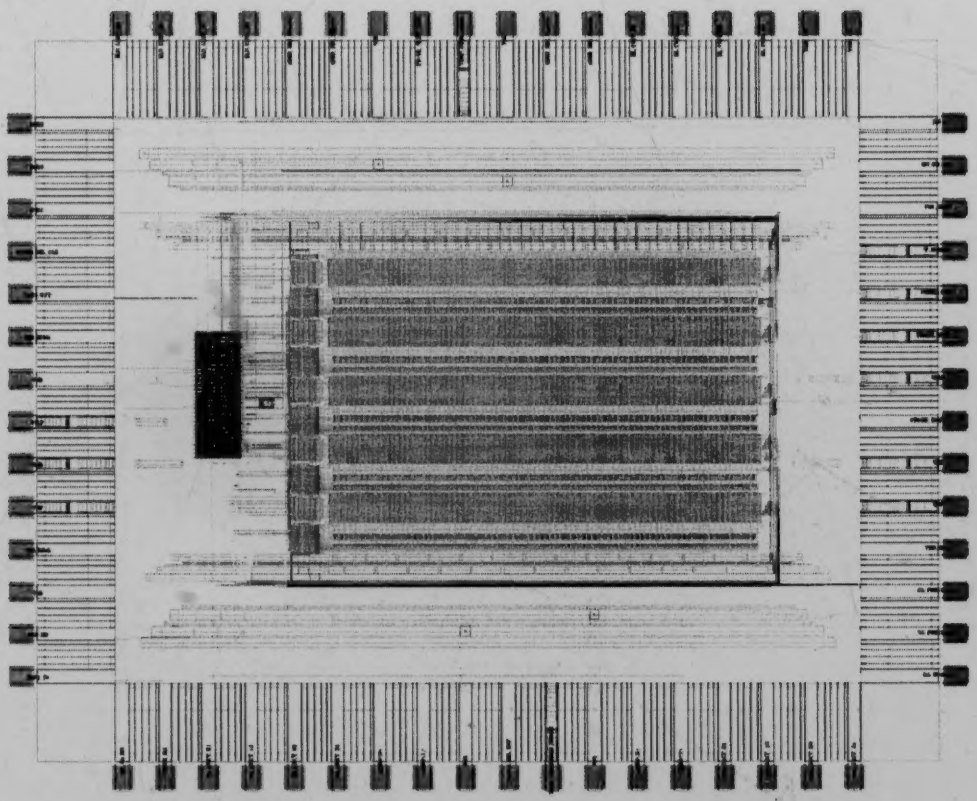


Figure C.1: ML6 die showing die pads, the core and the periphery

Appendix D

Sample Output from Visual C++ Code — tester output manipulation

```
Username= saung
Chipname= m16_1
Testname= functional
AmbTemp= 20
ChipTemp= 20
Comments= N/A
Date= 18/08/2003
Time= 17:45:22
Vbb= 1.000
Vblp= 0.900
Vpp= 2.500
Vdd= 1.800
Vdd_ring= 3.300
P_vref_in= 0.000
P_extv= 0.000
V_analog= 0.000
Vss= 0.000
CLK= 20
Load= 10000
Format= sec row col read write
```

```
0 0 0 00001 00000
0 0 0 00001 00001
0 0 0 00011 00011
0 0 0 00111 00111
0 0 0 11111 01111
0 0 0 11111 11111
0 0 1 00000 00000
0 0 1 00001 00001
0 0 1 00111 00011
0 0 1 00111 00111
0 0 1 01111 01111
0 0 1 11111 11111
0 0 2 00000 00000
0 0 2 00001 00001
0 0 2 00011 00011
0 0 2 00011 00111
0 0 2 00011 01111
0 0 2 11111 11111
0 0 3 00000 00000
0 0 3 00001 00001
0 0 3 00011 00011
0 0 3 00111 00111
0 0 3 01111 01111
0 0 3 11111 11111
0 0 4 00000 00000
0 0 4 00000 00001
0 0 4 00000 00011
0 0 4 00111 00111
0 0 4 01111 01111
0 0 4 01111 11111
```

Appendix E

Probe Point Table

Table E.1: ML6 Probe Points

No	Section	Description
0	E	BLn 319
1	E	BL 319
2	E	BLn 318
3	E	BL 318
4	E	BLn 317
5	E	BL 317
6	E	BLn 316
7	E	BL 316
8	E	BLn 315
9	E	BL 315
10	D	BLn 319
11	D	BL 319
12	D	BLn 318
13	D	BL 318
14	D	BLn 317
15	D	BL 317
16	D	BLn 316
17	D	BL 316
18	D	BLn 315
19	D	BL 315
20	C	BLn 319
21	C	BL 319
22	C	BLn 318
23	C	BL 318
24	C	BLn 317
25	C	BL 317

continued on next page

Table E.1: *continued*

No	Section	Description
26	C	BLn 316
27	C	BL 316
28	C	BLn 315
29	C	BL 315
30	B	BLn 319
31	B	BL 319
32	B	BLn 318
33	B	BL 318
34	B	BLn 317
35	B	BL 317
36	B	BLn 316
37	B	BL 316
38	B	BLn 315
39	B	BL 315
40	A	BLn 319
41	A	BL 319
42	A	BLn 318
43	A	BL 318
44	A	BLn 317
45	A	BL 317
46	A	BLn 316
47	A	BL 316
48	A	BLn 315
49	A	BL 315
50	-	DB 39
51	-	DBn 39
52	B	RWLn on BLn 318
53	B	RWL on BL 318
54	B	GWLn on BLn 318
55	B	GWL on BL 318
56	B	WL 1 on BLn 317
57	D	Reference cell - RWLn, BL 317
58	D	Reference cell - RWL, BLn 317
59	B	Sense Amplifier - BL 315
60	B	Sense Amplifier - BLn 315
61	B	Sense Amplifier - BL 318
62	B	Sense Amplifier - BLn 318
63	-	Selects probe_extv

Appendix F

Row Access Waveforms - the RGX Waveforms

Row access waveforms - the RGX waveforms

As in ML3 and ML5, the RGX signals are used in accessing the wordlines for each section. RGX stands for Reference Generate X (as in row). The reference and generate wordlines are turned on and off for each section according to the wordline that is selected.

The waveform patterns depend on whether the even or odd wordline is accessed and the section the wordline is accessing.

With this understanding of the three general waveform types, a decoder can be created to create this waveforms according to the wordline type (true or complement) and the section it is accessing.

The x addresses are first decoded into the respective reference and generate signals for each section. Depending on the sections that are addressed, the x-decoder is enabled so that the appropriate reference and generate signals are generated for the row access of the particular section.

Using the RGX decoders with the x decoders enables the saving of input pins for the reference and generate signals. The reference and generate input pins are replaced with the 3 RGX signal input pins on the MLDRAM chip.

Appendix G

PERL program used for cell yield calculations

```
#!/usr/bin/perl
##############################################################
# cellsizes.pl
#
# Author:
#   Ung
#   John Koob
#
# Date:
#   2003/10/16
#
# Description:
#
# Usage:
#   see &Usage
#
##############################################################

use Getopt::Std;
use File::Basename;

$0 = basename($0);
local($raw_file);
local($level);
local($size);
local($all);
local($sense_amp) = 0;
local($shielding) = 0;
local($max_size) = 4;
local($bitlines) = 80;
```

```
# no. bitlines in block w/ cells of size $size


&Usage(1) if (!&getopts('r:l:s:am:d:h'));
$opt_h && &Usage(0);

&Usage(1) if (! $opt_r);
&Usage(1) if (! $opt_l);

$raw_file = $opt_r;
$level = $opt_l;
$size = $opt_s;
$all = $opt_a;
$sense_amp = $opt_m;
$shielding = $opt_d;

if (! -e "$raw_file")
{
    &ReportError("$0: $raw_file does not exist\n",
    $verbose);
}

if ($all)
{
    &CellSizesAll($raw_file, $level, $max_size,
    $bitlines, $sense_amp, $shielding);
}
else
{
    &CellSizes($raw_file, \*STDOUT, $level, $size,
    $bitlines, $sense_amp, $shielding);
}

exit(0);


##########################################################
# CellSizesAll
#
# Desc:
#   Parse data for a given cell size.
#   Data block size is different for a different level.
#
# Input:
#
# Output:
```

```
#
############################################################
sub CellSizesAll
{
    my($raw_file, $level, $max_size, $bitlines,
$sense_amp, $shielding) = @_;
    my($i);
    my($out_file);
    my($tag);

    if ($sense_amp)
    {
$tag = "sa${sense_amp}_";
    }
    elsif ($shielding)
    {
$tag = "shield${shielding}_";
    }
    else
    {
$tag = "cell";
    }

    for ($i = 0; $i < $max_size; $i++)
    {
print "Extracting data for size $i ...\n";

($out_file =
$raw_file) =~ s/(\w+)\.(\w+)/$1_$tag$i.$2/;
open(OUT_FH, ">$out_file") ||
    &ReportError("$0: Cannot open $out_file\n",
    $verbose);

&CellSizes($raw_file, \*OUT_FH, $level, $i,
$bitlines, $sense_amp, $shielding);

close(OUT_FH);

print
"Calculating yield table for size $i ...\n";

($yld_file =
$out_file) =~ s/(\w+)\.(\w+)/$1.dat/;
`./wiggy.pl -r $out_file > $yld_file`;

print "\n";
```

```
    }
}

#############################################################
# CellSizes
#
# Desc:
#   Parse data for a given cell size.
#   Data block size is different for a different level.
#
# Input:
#
# Output:
#
#############################################################
sub CellSizes
{
    my($raw_file, $out_fh, $level, $size, $bitlines,
$sense_amp, $shielding) = @_;
    my(@lines);
    my($line);
    my($data) = 0;   # raw data line count
    my($count) = 0;  # cell count
    my($max_size) = 4;
    my($verbose) = 1;
    my($offset);

    (open(FH, "$raw_file") && (@lines=<FH>) && close(FH))
|| &ReportError("$0: Cannot open $raw_file\n",
$verbose);

    chomp(@lines);
    chop(@lines);

    foreach $line (@lines)
    {
if ((($line =~ /^\s*#/) || ($line =~ /=/)) ||
    ($line =~ /^\s*$/) || ($line =~ /320/))
{
    # next if ($line =~ /320/);
    # print "$line\n";
    next;
}

$index =
$size + $max_size * (int $count / ($max_size * $bitlines));
```

```
if ($sense_amp)
{
    $offset = ($sense_amp - 1)/2;
    if (($count >= $bitlines * ($index + $offset)) &&
($count < $bitlines * ($index + $offset + 0.5)))
    {
print $out_fh "$line\n";
    }
}
elsif ($shielding)
{
    $offset = ($shielding - 1)/4;
    if (($count >= $bitlines * ($index + $offset)) &&
($count < $bitlines * ($index + $offset + 0.25)))
    {
print $out_fh "$line\n";
    }
    if (($count >= $bitlines * ($index + $offset + 0.5)) &&
($count < $bitlines * ($index + $offset + 0.75)))
    {
print $out_fh "$line\n";
    }
}
else
{
    if (($count >= $bitlines * ($index)) &&
($count < $bitlines * ($index + 1)))
    {
print $out_fh "$line\n";
    }
}

$data++;
$count = int $data / $level;
    }
}

##########################################################
# ReportError
#
# Desc:
#   Print error message.
#
# Input:
#   $message    - error message
```

```
#   $display    - flag to control display of error message
#
# Output:
#   none
#
#############################################################
sub ReportError
{
my($message, $display) = @_;

print STDERR $message if ($display);
exit(1);
} # ReportError


#############################################################
# Usage
#
# Desc:
#   Print usage statement.
#
# Input:
#   $ret_code    - program exit code
#
# Output:
#   exits with $ret_code
#
#############################################################
sub Usage
{
my($ret_code) = @_;

print "\nUsage: $0 -r <raw_file> -l <level> -s <cellsize> ";
print "          [ -a ] [ -m ] [ -d ] [ -h ] ";
print "\nwhere\n";
print " -r raw benchmark data file\n";
print " -l number of voltage levels\n";
print " -s size of cell (tiny=0,small=1,med=2,large=3)\n";
print " -a all four cell sizes \n";
print " -m number of sense amp (1 or 2)\n";
print " -d shielding (1) or no shielding (2)\n";
print "  \n";
print "  \n";

exit $ret_code;
} # Usage
```